UFR Sciences
M2 Calcul Formel
Option Maths approfondies

2012–2013
M. WEIMANN

Factorisation dans $\mathbb{F}_p[x]$

La factorisation des polynômes est un des problèmes fondamentaux du calcul formel, sur lequel beaucoup de progrès significatifs ont été faits ces 30 dernières années. Les situations et les problèmes soulevés sont variés : une ou plusieurs variable, factorisation dans un anneau ou dans un corps, factorisation rationnelle ou absolue, polynômes denses, convexe-denses, creux, complexité arithmétique ou complexité en bits, gestion mémoire, etc. Les techniques pour approcher ces problèmes sont variées (corps finis, algèbre linéaire, théorie des nombres, géométrie algébrique, géométrie diophantienne, algorithmique, etc.)

Un des problèmes fondamentaux est la factorisation des polynômes univariés à coefficients dans un corps fini $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$, où p est un nombre premier. Ce problème à de nombreuses applications, il est entre autres à la base de nombreux autres algorithmes de factorisation, en particulier la factorisation sur $\mathbb{Z}[x]$ que nous étudierons au chapitre suivant. **Schéma...**

Comme l'anneau $\mathbb{F}_p[x]$ est factoriel, tout polynôme univarié $f \in \mathbb{F}_p[x]$ se factorise de manière unique sous la forme

$$f = c f_1^{m_1} \cdots f_s^{m_s},$$

où $c \in \mathbb{F}_p^*$, $m_i \in \mathbb{N}^*$, et les $f_i \in \mathbb{F}_p[x]$ sont des polynômes irréductibles unitaires non constants et distincts deux à deux.

L'objectif de ce chapitre est de développer un algorithme qui, étant donné $f \in \mathbb{F}_p[x]$, retourne la liste $[c, (f_1, m_1), \ldots, (f_s, m_s)]$. On étudiera dans un premier temps le cas réduit, ou sans facteurs multiples ($m_i = 1$), puis dans un second temps le cas général.

On suggère vivement comme références très complètes sur les corps finis puis sur le calcul formel les deux "bibles" :

- Lidl and Niederreiter, *Finite fields*, Encyclopedia of Mathematics and its applications 20, Cambridge University Press (1997).
- Gathen and Gerhard, *Modern Computer Algebra*, Cambridge University Press (2003).

1 Le cas sans facteurs multiples

On suppose dans cette section que f est sans facteurs multiples ($m_i = 1$ pour tout i). On va étudier deux des algorithmes de factorisation les plus répandus dans les logiciels de calculs formels : l'algorithme de Berlekamp et l'algorithme de Cantor-Zassenhaus. Il existe de nombreux autres algorithmes plus ou moins efficaces selon les situations.

1.1 La méthode de Berlekamp

La méthode de Berlekamp permet de ramener à un problème d'algèbre linéaire le calcul du nombre de facteurs irréductibles d'un polynôme $f \in \mathbb{F}_p[x]$ sans facteurs carrés.

1.1.1 Trouver le nombre de facteurs

L'objet central de l'algorithme de Berlekamp est l'algèbre quotient

$$R = \mathbb{F}_p[x]/(f).$$

C'est une $\mathbb{F}_p$-algèbre de rang $d := deg(f)$. Comme les f_i sont irréductibles, les algèbres quotients

$$R_i = \mathbb{F}_p[x]/(f_i)$$

sont des corps, extensions finies de $\mathbb{F}_p$ de degrés $\deg(f_i)$. Les f_i étant premiers deux à deux, le lemme des restes chinois nous assure que l'application

$$\begin{array}{rccc} r: & R & \longrightarrow & R_1 \times \cdots \times R_s \\ & g \bmod f & \longmapsto & (g \bmod f_1, \ldots, g \bmod f_s) \end{array} \tag{1}$$

est un isomorphisme d'algèbres. Nous noterons r_i les applications coordonnées, $r = (r_1, \ldots, r_s)$.

Par ailleurs, l'application de Frobenius ϕ_p, définie par

$$\begin{array}{rccc} \phi_p: & R & \to & R \\ & g & \mapsto & g^p \end{array} \tag{2}$$

est $\mathbb{F}_p$-linéaire. En effet, l'algèbre R étant de caractéristique p, on a

$$(f+g)^p = f^p + g^p,$$

puisque, si p est un nombre premier, les coefficients binomiaux $\binom{j}{p}$, C_p^j pour $1 \leq j \leq p-1$ sont multiples de p.

On appelle *la sous-algèbre de Berlekamp* $\mathcal{B} \subset R$ l'ensemble des points fixes de ϕ_p. Autrement dit,

$$\mathcal{B} := \ker(\phi_p - Id),$$

où Id est l'application identité de R. Le morphisme $r = (r_1, \ldots, r_s)$ étant un isomorphisme d'algèbres *compatible* avec les actions des Frobenius, on a les équivalences

$$g \in \mathcal{B} \iff r_i(g)^p = r_i(g)\ \forall i.$$

Rappelons le lemme bien connu suivant :

Lemme 1 (petit théorème de Fermat) Soit K une extension finie de $\mathbb{F}_p$. Alors $\mathbb{F}_p$ est le sous-ensemble des points de K fixés par le Frobenius : $\mathbb{F}_p = \{z \in K,\ z^p = z\}$.

Preuve. Soit $z \in \mathbb{F}_p$. Si $z = 0$, il est clair que $z^p = z$. Sinon $z \in \mathbb{F}_p \setminus \{0\}$, ensemble qui coïncide avec le groupe multiplicatif $\mathbb{F}_p^*$ des inversibles de $\mathbb{F}_p$ (car $\mathbb{F}_p$ est un corps). Ce groupe fini étant de cardinal $p-1$, l'ordre de z divise $p-1$. Ainsi $z^{p-1} = 1$ et $z^p = z$. Le sous-ensemble de K constitué

des solutions de $z^p - z = 0$ étant de cardinal au plus p (un polynôme de degré p a au plus p racines), et qu'il contient $\mathbb{F}_p$, c'est donc $\mathbb{F}_p$. □

Ainsi, $g \in \mathcal{B}$ si et seulement si $r_i(g) \in \mathbb{F}_p$, pour tout $i = 1, \ldots, s$. En particulier, r induit un isomorphisme

$$r : \mathcal{B} \longrightarrow \mathbb{F}_p^s$$

et $\mathcal{B}$ est un espace vectoriel sur $\mathbb{F}_p$ dont la dimension coïncide avec le nombre s de facteur irréductibles de f. Nous avons donc un moyen de déterminer s en calculant la dimension du noyau de la matrice de l'endomorphisme $\phi_p - Id$ (en pratique, on utilisera la base $(1, x, \ldots, x^{n-1})$ de R).

1.1.2 Déterminer les facteurs : approche déterministe

Voyons maintenant comment déterminer effectivement ces facteurs, si $s > 1$. D'aprés le Lemme 1, on a dans $\mathbb{F}_p[x]$ l'égalité :

$$x^p - x = \prod_{\omega \in \mathbb{F}_p} (x - \omega) \tag{3}$$

de sorte que pour tout $g \in \mathbb{F}_p[x]$ avec $\deg(g) \geq 1$ on a par composition :

$$g^p - g = \prod_{\omega \in \mathbb{F}_p} (g - \omega).$$

Les facteurs $g - \omega$ étant premiers deux à deux, on a :

$$\mathrm{pgcd}(f, g^p - g) = \prod_{\omega \in \mathbb{F}_p} \mathrm{pgcd}(f, g - \omega). \tag{4}$$

En particulier, si $\bar{g} := g \mod (f)$ appartient à $\mathcal{B}$, on a $\mathrm{pgcd}(f, g^p - g) = f$ et (4) donne une factorisation de f :

$$f = \prod_{\omega \in \mathbb{F}_p} \mathrm{pgcd}(f, g - \omega). \tag{5}$$

Le lemme suivant nous assure que cette factorisation n'est pas triviale dès lors que $\bar{g} \in R \setminus \mathbb{F}_p$ (g non constant modulo (f)).

Lemme 2 Pour tout $\bar{g} \in \mathcal{B}$ non constant, il existe $\omega \in \mathbb{F}_p$ tel que $\mathrm{pgcd}(f, g - \omega)$ soit un facteur non trivial de f (différent de 1 et f).

Preuve. D'après (5), il existe ω tel que $\mathrm{pgcd}(f, g - \omega)$ soit un diviseur de f non constant. Comme g est non constant modulo (f), on a forcément $\mathrm{pgcd}(f, g - \omega) \neq f$, donc $\mathrm{pgcd}(f, g - \omega)$ est un facteur non trivial de f. □

Ainsi, si $s > 1$, les facteurs de f sont à rechercher parmis la liste des polynômes $\mathrm{pgcd}(f, g - \omega)$, où g parcourt une base de $\mathcal{B}$ et $\omega \in \mathbb{F}_p$. En partique, on cherche une factorisation non triviale $f = f_1 f_2$ et on rappelle récursivement l'algorithme sur f_1 et f_2.

1.1.3 Déterminer les facteurs : approche probabiliste

L'approche précédante requiert de calculer $\mathrm{pgcd}(f, g-\omega)$ pour g parcourant une base de $\mathcal{B}$ et ω parcourant $\mathbb{F}_p$. La complexité est au moins linéaire en p et l'algorithme s'avère vite impraticable lorsque p est grand. Il existe une autre approche probabiliste plus performante pour p grand.

Plutôt que de chercher les facteurs de f parmis les facteurs $g-\omega$ de g^p-g, nous allons utiliser la factorisation

$$g^p - g = g(g^{\frac{p-1}{2}} - 1)(g^{\frac{p-1}{2}} + 1) \tag{6}$$

valable pour tout premier $p \neq 2$. On a alors $\bar{g} \in \mathcal{B}$ si et seulement si f divise $g^p - g$, si et seulement si

$$f = \mathrm{pgcd}(f, g^p - g) = \mathrm{pgcd}(f, g)\,\mathrm{pgcd}(f, g^{\frac{p-1}{2}} - 1)\,\mathrm{pgcd}(f, g^{\frac{p-1}{2}} + 1).$$

Si $\bar{g} \neq 0$ et $g_1 := \mathrm{pgcd}(f, g) \neq 1$, alors on a trouvé un facteur g_1 non trivial de f et on rappelle l'algorithme récursivement sur g_1 et f/g_1. Si $g_1 = 1$, alors $g_2 := \mathrm{pgcd}(f, g^{\frac{p-1}{2}} - 1) = f$ si et seulement si

$$g^{\frac{p-1}{2}} - 1 \equiv 0 \mod (f_i) \quad \forall\, i = 1, \ldots, s \tag{7}$$

et $g_2 = 1$ si et seulement si $g_3 := \mathrm{pgcd}(f, g^{\frac{p-1}{2}} + 1) = f$, si et seulement si

$$g^{\frac{p-1}{2}} + 1 \equiv 0 \mod (f_i) \quad \forall\, i = 1, \ldots, s. \tag{8}$$

Le lemme suivant montre que ces deux situations de factorisation triviale $g_2 \in \{1, f\}$ n'arrivent qu'avec faible probabilité :

Lemme 3 Soit $p \neq 2$ un nombre premier. Les deux sous-ensembles

$$S_+ := \{z \in \mathbb{F}_p^*, \quad z^{\frac{p-1}{2}} - 1 = 0\} \quad \text{et} \quad S_- := \{z \in \mathbb{F}_p^*, \quad z^{\frac{p-1}{2}} + 1 = 0\}$$

sont de même cardinal $(p-1)/2$ et réalisent une partition de $\mathbb{F}_p^*$.

Preuve. Par le petit théorème de Fermat (Lemme 1), on a

$$0 = z^{p-1} - 1 = (z^{\frac{p-1}{2}} - 1)(z^{\frac{p-1}{2}} + 1) \quad \forall\, z \in \mathbb{F}_p^*.$$

la deuxième égalité utilisant l'hypothèse p impair. On a donc une partition $\mathbb{F}_p^* = S_+ \cup S_-$ et il s'ensuit que

$$\mathrm{Card}(S_+) + \mathrm{Card}(S_-) = \mathrm{Card}(\mathbb{F}_p^*) = p - 1. \tag{9}$$

D'autre part, les polynômes $x^{(p-1)/2} - 1$ et $x^{(p-1)/2} + 1$ de $\mathbb{F}_p[x]$ ont chacun au plus $(p-1)/2$ racines et ainsi

$$\mathrm{Card}(S_+) \leq \frac{p-1}{2} \quad \text{et} \quad \mathrm{Card}(S_-) \leq \frac{p-1}{2}. \tag{10}$$

En combinant (9) et (10), on obtient le résultat. □

Rappelons que $\bar{g} \in \mathcal{B}$ si et seulement si $g \mod (f_i) \in \mathbb{F}_p \subset \mathbb{F}_p[x]/(f_i)$ pour tout $i \in \mathbb{F}_p^*$. Ainsi, si $\bar{g} \in \mathcal{B}$ est combinaison linéaire des éléments d'une base de $\mathcal{B}$ à coefficients dans $\mathbb{F}_p$ aléatoires, alors $g \mod (f_i) \in \mathbb{F}_p$ prend chaque valeur de $\mathbb{F}_p$ avec équiprobabilité $1/p$, et ce pour $i = 1, \ldots, s$. Il suit du lemme précédant, que si $g \mod (f_i) \neq 0$, alors les deux évènements $g^{(p-1)/2} - 1 = 0 \mod f_i$ et $g^{(p-1)/2} + 1 = 0 \mod f_i$ ont équiprobabilité $1/2$. Ainsi, chaque évènement (7) et (8) apparaît avec

probabilité $1/2^s$. Donc g_2 est un facteur trivial de f avec probabilité $1/2^{(s-1)}$, quantité inférieure ou égale à $1/2$ dés lors que $s \geq 2$. Si g_2 est un facteur trivial, on recommence avec un autre choix de g. La probabilité de n'obtenir que des facteurs triviaux après k itérations vaut

$$1/2^{k(s-1)} \leq 1/2^k,$$

et la probabilité de trouver une factorisation non triviale de f tend vers 1 lorsque k tend vers l'infini. En pratique, 1 ou 2 itérations suffisent, et l'approche probabiliste aura une meilleure complexité en moyenne que l'approche déterministe (cf exercices).

1.2 La méthode de Cantor-Zassenhaus

Nous avons vu la méthode de Berlekamp, essentiellement basée sur l'algèbre linéaire. On va maintenant voir une autre méthode connue et fréquemment utilisée, la méthode de Cantor-Zassenhaus. Elle consiste dans un premier temps à regrouper les facteurs de même degrés (*distinct degree factorization*) puis à casser ces groupes de facteurs selon une méthode probabiliste similaire à celle développée ci-dessus. Commençons par une remarque simple sur la recherche de racines.

1.2.1 Recherche de racines

L'algorithme précédant admet comme application immédiate la recherche des racines rationnelles (c'est à dire dans $\mathbb{F}_p$) d'un polynôme $f \in \mathbb{F}_p[x]$ sans facteurs multiples. En effet, ces racines sont en bijection avec les facteurs irréductibles de degrés 1 de f. Il suffit donc de factoriser f sur $\mathbb{F}_p[x]$ et de chercher les facteurs de degrés 1. Cependant, il n'est pas satisfaisant d'un point de vue complexité de factoriser complètement un polynôme si l'on ne recherche que les facteurs d'un degré donné. Le lemme 1 permet de résoudre simplement ce problème. En effet, on sait que le polynôme $x^p - x$ est le produit de tous les polynômes (unitaires) de degrés 1 :

$$x^p - x = \prod_{\omega \in \mathbb{F}_p} (x - \omega). \tag{11}$$

Ainsi,

$$h := \text{pgcd}(x^p - x, f)$$

est égal au produit de tous les facteurs de degré 1 de f, et il suffit d'appliquer l'algorithme de Berlekamp sur h pour trouver les facteurs de degrés 1 (donc les racines) de f. On va maintenant voir comment généraliser cette méthode à la recherche de facteurs irréductibles de degrés fixés.

1.2.2 Distinct Degree Factorization

Plus généralement, on peut factoriser un polynôme $f \in \mathbb{F}_p[x]$ en groupes de facteurs irréductibles de même degré (distinct degree factorization). Cette approche est basée sur la théorie des corps finis. Rappellons que tout polynôme $f \in \mathbb{F}_p[x]$ irréductible de degré d définit une extension de degré d de $\mathbb{F}_p$

$$K := \mathbb{F}_p[x]/(f).$$

K est un corps fini à p^d éléments, et l'ensemble des inversibles $K^* = K \setminus \{0\}$ est un groupe multiplicatif de cardinal $p^d - 1$. Ainsi, tout élément $z \in K^*$ est racine du polynôme $x^{p^d-1} - 1$, et tout élément de K est racine du polynôme

$$Q_d := x^{p^d} - x.$$

Comme Q_d a au plus p^d racines, il s'ensuit que

$$K = \{z \in \bar{\mathbb{F}}_p,\ z^{p^d} - z = 0\},$$

où $\bar{\mathbb{F}}_p$ désigne la clôture intégrale de $\mathbb{F}_p$. Ceci montre qu'à isomorphisme près, il y a une unique extension de degré d de $\mathbb{F}_p$. On la note $\mathbb{F}_q$, avec $q = p^d$. Comme pour le cas $d = 1$, on a alors l'égalité

$$x^{p^d} - x = \prod_{z \in \mathbb{F}_q} (x - z)$$

dans $\mathbb{F}_q[x]$. Le théorème suivant donne une bonne description des polynômes irréductibles de $\mathbb{F}_p[x]$:

Théorème 1 Pour tout $d \geq 1$, le polynôme $Q_d(x) := x^{p^d} - x \in \mathbb{F}_p[x]$ est le produit de tous les polynômes unitaires irréductibles dont le degré divise d.

Preuve. Comme $Q'_d = -1$, on a $\text{pgcd}(Q_d, Q'_d) = 1$ et Q_d est sans facteurs multiples (voir section suivante). Il suffit donc de montrer que si f est unitaire irréductible de degré n, on a l'équivalence

$$f \quad \text{divise} \quad Q_d \iff n \quad \text{divise} \quad d.$$

D'après ce qui précède, tout polynôme f irréductible unitaire de degré n définit une extension $K_n := \mathbb{F}_{p^n}$ de degré n de $\mathbb{F}_p$, qui est le plus petit corps contenant toutes les racines de f. Si f divise Q_d dans $\mathbb{F}_p[x]$, alors toute racine de f est racine de Q_d. Comme le corps $\mathbb{F}_{p^d}$ est l'ensemble des racines de Q_d, on a une inclusion de corps

$$\mathbb{F}_{p^n} \subset \mathbb{F}_{p^d}.$$

Il suit que $\mathbb{F}_{p^d}$ est une extension de $\mathbb{F}_{p^n}$, donc un $\mathbb{F}_{p^n}$-espace vectoriel. Un argument de cardinalité implique que $p^d = (p^n)^e$ pour un certain entier e et n divise p. Inversement, supposons que n divise d. Si $f \in \mathbb{F}_p[x]$ est unitaire irréductible de degré n, il se factorise totalement dans $\mathbb{F}_{p^n}[x]$:

$$f(x) = \prod_{z \in \mathbb{F}_{p^n}, f(z)=0} (x - z)$$

et donc f divise Q_n dans $\mathbb{F}_{p^n}[x]$, donc f divise Q_n dans $\mathbb{F}_p[x]$ car f et Q_n sont tous les deux à coefficients dans $\mathbb{F}_p$. Comme n divise d, $p^n - 1$ divise $p^d - 1$ donc Q_n divise Q_d (utiliser l'argument $x - 1$ divise $x^k - 1$ pour tout $k \in \mathbb{N}$). Ainsi, f divise Q_d quand n divise d. □

On déduit rapidement de ce théorème un algorithme **DistDegFact**(f, p) qui renvoie la liste des groupes de facteurs irréductibles de même degrés. Attention, penser à utiliser un algorithme d'exponentiation rapide dans $\mathbb{F}_p[x]/(f)$ pour calculer les différentes puissances mises en jeux.

1.2.3 Cantor-Zassenhaus

Puisque l'on peut facilement écrire un polynôme sans facteurs carrés comme produit de facteurs irréductibles de même degré, il nous suffit maintenant de savoir factoriser de tels polynômes. L'approche est similaire à celle développée pour Berlekamp probabiliste.

Soit donc un polynôme $f \in \mathbb{F}_p[x]$ sans facteur carré, produit de facteurs irréductibles de degré exactement r. D'après la section précédente, on sait donc que P divise $x^{p^r} - x$. D'autre part, on a le lemme suivant

Lemme 4 Pour tout $g \in \mathbb{F}_p[x]$, $x^{p^r} - x$ divise $g^{p^r} - g$.

Preuve. Soit $g = \sum a_i x^i$. Puisque l'on travaille sur $\mathbb{F}_p[x]$, on a $g^{p^r}(x) = g(x^{p^r})$ et

$$g^{p^r}(x) - g(x) = g(x^{p^r}) - g(x) = \sum a_i (x^{ip^r} - x^i)$$

est divisible par $x^{p^r} - x$ car chacun des termes $x^{ip^r} - x^i$ l'est. □

On écrit maintenant

$$g^{p^r} - g = g(g^{(p^r-1)/2} - 1)(g^{(p^r-1)/2} + 1)$$

Ces trois facteurs étant premiers entre eux, et comme f divise $g^{p^r} - g$, on obtient la d écomposition de f suivante :

$$f = \mathrm{pgcd}(f, g)\,\mathrm{pgcd}(f, g^{(p^r-1)/2} - 1)\,\mathrm{pgcd}(f, g^{(p^r-1)/2} + 1) \tag{12}$$

L'algorithme de Cantor-Zassenhaus consiste à tirer au sort un polynôme g de degré $< 2r$ et de vérifier si la décomposition précédente est non triviale. On a lemme suivant

Lemme 5 Soit $f \in \mathbb{F}_p[x]$ sans facteurs carrés produit d'au moins 2 facteurs irréductibles de même degrés r. Alors la probabilité que la factorisation (12) induite par $g \in \mathbb{F}_p[x]$ de degré $< 2r$ soit trivial est d'au moins $1/2$.

Preuve. La preuve est dans le même esprit que la situation décrite pour Berlekamp probabiliste.

Le principe de l'algorithme est donc simple : choisir g aléatoire jusqu'à ce que (12) donne une factorisation non trivial de f. Attention, penser à utiliser un algorithme d'exponentiation rapide dans $\mathbb{F}_p[x]/(f)$ pour calculer g^{p^r-1}.

2 Le cas avec facteurs multiples

Si un polynôme $f \in \mathbb{F}_p[x]$ a des facteurs multiples, alors le résultant de f et de sa dérivée s'annule :

$$Res(f, f') = 0.$$

En effet, si g^2 divise f, alors g divise f' : les facteurs multiples sont des facteurs communs à f et f' et sont donc à rechercher parmis les facteurs de $\mathrm{pgcd}(f, f')$. La dérivée de f va donc jouer un rôle central ici. Contrairement aux corps de caractéristique nulle, une difficulté s'ajoute : un polynôme

peut être de dérivée nulle sans être constant (par exemple $f(x) = x^2$ dans $\mathbb{F}_2[x]$). Ces polynômes sont caractérisés par le lemme suivant.

Lemme 6 Soit $f \in \mathbb{F}_p[x]$ un polynôme unitaire non constant. On a $f' = 0$ si et seulement s'il existe $q \in \{p, p^2, p^3, \ldots\}$ et $g \in \mathbb{F}_p[x]$ tels que $f = g^q$ et $g' \neq 0$.

Preuve. Si $f = g^q$ pour q une puissance de p, alors $f' = qg^{q-1} = 0$ car p divise q. Supposons maintenant $f' = 0$. Ecrivons $f = \sum_{i=0}^{n} c_i x^i$. Alors $f' = \sum_{i=0}^{n} i c_i x^i$ et $f' = 0$ implique $ic_i = 0$ pour $i = 0, \ldots, n$. Ainsi $c_i = 0$ pour tout i non nul modulo p et

$$f(x) = c_0 + c_p x^p + \cdots + c_{kp} x^{kp} = (c_0 + c_p x + \cdots + c_{kp} x^k)^p.$$

La deuxième égalité est basée sur la linéarité du Frobenius : $(cg + dh)^p = cg^p + dh^p$ pour tout h, g dans $\mathbb{F}_p[x]$ et tout c, d dans $\mathbb{F}_p$. Ainsi $f = g_0^p$ pour un certain $g_0 \in \mathbb{F}_p[x]$ non constant. Si $g_0' = 0$, on a par le même raisonnement $g_0 = g_1^p$ et $f = g_1^{p^2}$. Si $g_1' = 0$, on continue...Ce processus s'arrête et $f = g^q$ pour un $q \in \{p, p^2, p^3, \ldots\}$ inférieur ou égal à $\deg(f)$. On a $g' \neq 0$ car q est maximal et g est non constant. □

Corollaire 1 Soit $f \in \mathbb{F}_p[x]$ un polynôme unitaire non constant. Alors f a un facteur multiple si et seulement si $Res(f, f') = 0$.

Preuve. On a vu que si g^2 divise f alors $Res(f, f') = 0$. Supposons maintenant que $Res(f, f') = 0$. Il existe donc un facteur commun irrédutible non constant $g \in \mathbb{F}_p[x]$ à f et f'. Supposons que g aie multiplicité 1 dans f. On a alors $f = gh$ avec $\mathrm{pgcd}(h, g) = 1$. On a $f' = g'h + h'g$. Comme par hypothèse g divise f et f', il suit que g divise $g'h$. Comme g et h sont supposés premiers entre eux, il suit que g divise g'. Comme $\deg(g') < \deg(g)$, on a forcément $g' = 0$, contredisant l'hypothèse g non constant irréductible par le lemme précédant. Donc g est un facteur multiple de f. □

Remarque Le lemme et son corollaire ne sont pas vrais sur n'importe quel corps de caractéristique p : on a toujours $f' = 0$ implique $f(x) = g(x^q)$, mais l'égalité $g(x^q) = g(x)^q$ n'est pas toujours satisfaite (penser par exemple $f(x) := x^p - t \in \mathbb{F}_p(t)[x]$. Ce polynôme non constant satisfait $f' = 0$ (et $\mathrm{pgcd}(f, f') = 0$), mais f est malgré cela irréductible sur le corps $\mathbb{F}_p(T)$. La différence réside dans le fait que le corps des coefficients $\mathbb{F}_p(t)$ n'est pas fixé par le Frobenius $f \to f^p$.

Les facteurs irréductibles de f apparaissant avec une multiplicité divisible par p sont donc à isoler des autres facteurs de f. On s'intéresse donc à factoriser f sous la forme

$$f = a \times b$$

avec $a = g^q$ pour q une puissance de p, b sans facteur irréductible de multiplicité divisible de p, et a et b premiers entre eux. Comment obtenir une telle factorisation ?

Etape 1. *On cherche b.* Ecrivons

$$b = b_1^{n_1} \times \cdots \times b_k^{n_k}$$

la décomposition irréductible de b. On a

$$b' = h \prod_{i=1}^{k} b_i^{n_i - 1}, \quad \text{avec} \quad h = \sum_{i=1}^{k} n_i b_i' \prod_{j \neq i} b_j.$$

Par hypothèse, b_i est premier avec b_j pour $j \neq i$. De plus, b_i étant irréductible il est premier avec b_i' (sinon, on aurait $b_i' = 0$ contredisant l'irréductibilité de b_i par le lemme précédant). Comme n_i est inversible modulo p par hypothèse, il en découle que

$$\mathrm{pgcd}(b_i, h) = \mathrm{pgcd}(b_i, n_i b_i' \prod_{j \neq i} b_j) = 1.$$

Il s'ensuit que $\mathrm{pgcd}(b, b') = \prod_{i=1}^{r} b_i^{n_i - 1}$ et

$$bb := \frac{b}{\mathrm{pgcd}(b, b')} = b_1 \times \cdots \times b_k$$

représente la partie sans facteurs carrés de b, que l'on peut factoriser grâce à l'algorithme de Berlekamp. Pour trouver bb à partir de f, il suffit de remarquer que $a' = 0$ entraine $f' = ab'$ et

$$\mathrm{pgcd}(f, f') = \mathrm{pgcd}(ab, ab') = a\,\mathrm{pgcd}(b, b').$$

Ainsi, bb s'obtient à partir de f comme

$$bb = \frac{b}{\mathrm{pgcd}(b, b')} = \frac{f}{\mathrm{pgcd}(f, f')}.$$

Finalement, la multiplicité n_i de b_i se calcule aisément, par exemple comme le plus petit entier m tel que b_i^m divise f. Ceci nous donne le facteur b de f et sa factorisation irréductible.

Etape 2. *On traite a.* Une fois b connu, le facteur a s'obtient simplement comme le quotient de f par b. Si a est constant, c'est fini et on retourne la factorisation de b. Sinon, il existe d'après le Lemme 3 un unique $g \in \mathbb{F}_p[x]$ et un unique $q \in \{p, p^2, p^3, \ldots\}$ tels que $a = g^q$ et $g' \neq 0$. Il est facile de déterminer g et q (exercices). On appelle alors récursivement l'algorithme de factorisation sur g. On en déduit la factorisation irréductible de g, dont on déduit la factorisation irréductible de a (il suffit de multiplier les multiplicités des facteurs de g par q). Finalement, on déduit des factorisations de a et de b la factorisation de f.

3 L'algorithme complet de factorisation dans $\mathbb{F}_p[x]$

En combinant toutes les procédures décrites précédamment, on obtient finalement un algorithme complet de factorisation sur $\mathbb{F}_p[x]$. On suppose donnée une procédure **Facto(f,p)** (type Berlekamp ou Cantor-Zassenhaus) qui prend en entrée un premier p et un polynôme $F \in \mathbb{Z}[x]$ sans facteurs carrés et qui retourne la liste $[f_1, \ldots, f_r]$ des facteurs irréductibles de F modulo (p).

FactoModp(F,p) :

Entrée : $F \in \mathbb{Z}[x]$ et $p \in \mathbb{N}$ premier.

Sortie : $L = [c, (f_1, m_1), \ldots, (f_s, m_s)]$, avec c coefficient dominant de $f \mod (p)$, les $f_i \in \mathbb{F}_p[x]$ facteurs moniques irréductibles non constants de $f := F \mod (p)$ et m_i leurs multiplicités.

$f := F \mod (p)$.

Si f est constant :

retourner $L := [f]$.

Sinon :

$c :=$ coefficient dominant de f

$f := f/c$.

Si $Res(f, f') \neq 0$:

Calculer $[f_1, \ldots, f_r] =$ **Facto(f,p)**

Retourner $L := [c, (f_1, 1), \ldots, (f_r, 1)]$

Sinon :

Calculer $bb := f/\operatorname{pgcd}(f, f')$.

Si $\deg(bb) = 0$:

Poser $L_b := \{\emptyset\}$

Sinon :

Calculer $[b_1, b_2, \ldots] :=$ **Facto (bb,p)**

Calculer les multiplicités b_i des facteurs b_i de f.

En déduire b et $L_b := [c, (b_1, n_1), (b_2, n_2), \ldots]$

Calculer $a :=$quotient(f, b).

Si $\deg(a) = 0$:

Retourner $L := L_b$

Sinon :

Calculer (g, q) tel que $a = g^q$, avec $g' \neq 0$

Calculer $[(g_1, e_1), (g_2, e_2), \ldots] =$ **FactoModp(g,p)** (algorithme récursif)

Calculer $L_a := [(g_1, qe_1), (g_2, qe_2), \ldots]$

Retourner $L := L_a + L_b$ où $+$ désigne la concaténation.

Théorème 2 L'algorithme **Factorisation** termine et retourne la factorisation irréductible de f.

Preuve. A chaque appel récursif de la fonction **Factorisation**, le degré du polynôme g décroît strictement. Donc l'algorithme termine. Il retourne la factorisation irréductible de f d'après les résultats préalablement prouvés.

4 Test d'irréductibilité et probabilité d'irréducibilité dans $\mathbb{F}_p[x]$

Comment décider rapidement si un polynôme est irréductible ? Quelle est la probabilité qu'un polynôme $f \in \mathbb{F}_p[x]$ unitaire de degré donné d soit irréductible ? Comment fabriquer un polynôme irréductible ?

Ces questions d'irréductibilité des polynômes représentent des enjeux importants du Calcul Formel, dû entre autres au fait qu'un polynôme irréductible de grand degré permet de construire des corps finis de grand cardinal et de petite caractéristique, utiles par exemples pour les codes correcteurs.

Comme on peut s'y attendre, le Théorème 1 joue un rôle central dans ces questions.

4.1 Tests d'irréductibilité

Tout algorithme de factorisation peut bien entendu être utilisé comme test d'irréductibilité. On peut cependant accélérer les choses si l'on ne cherche pas à calculer les facteurs. On a le test d'irréductibilité suivant :

Proposition 1 Un polynôme $f \in \mathbb{F}_p[x]$ de degré $d \geq 1$ est irréductible si et seulement si

1. f divise $x^{p^d} - x$ et
2. $\mathrm{pgcd}(x^{p^e} - x, f) = 1$ pour tout diviseur premier e de d.

Preuve. C'est une application immédiate du Théorème 1. □.

4.2 Probabilité d'irréductibilité

Le cas $d = q$ premier. Notons $\tilde{Q}_q$ le produit de tous les polynômes unitaires irréductibles de degré q. Soit $h \in \mathbb{F}_p[x]$ irréductible divisant Q_q. On sait d'après le théorème 1 que $\deg(h)$ divise q. Comme q est premier, on a donc $\deg(h) = q$ ou $\deg(h) \leq 1$. On a donc

$$Q_q = Q_1 \times \tilde{Q}_q$$

Il s'ensuit que

$$\deg(\tilde{Q}_q) = \deg(Q_q) - \deg(Q_1) = p^q - p.$$

Il y a donc $(p^q - p)/q$ polynômes irréductibles unitaires de degré q parmi les p^q polynômes unitaires de degré q. Un polynôme unitaire de degré q a donc une probabilité

$$\frac{p^q - p}{qp^q} = \frac{1}{q}\Big(1 - \frac{1}{p^{q-1}}\Big)$$

d'être irréductible. Par exemple, un polynôme de degré 2 dans $\mathbb{F}_2[x]$ a une probabilité $1/4$ d'être irréductible (il n'y en a qu'un seul, c'est le polynôme $x^2 + x + 1$ parmis les 4 polynômes unitaires $x^2 + x + 1$, $x^2 + x$, $x^2 + 1$, x^2 de $\mathbb{F}_2[x]$ de degré 2).

Le cas $d = q^n$, q premier. Soit $h \in \mathbb{F}_p[x]$ irréductible divisant Q_{q^n}. On sait d'après le théorème 1 que $\deg(h)$ divise q^n. Comme q est premier, soit $\deg(h) = q^n$ (et h divise $\tilde{Q}_{q^n}$), soit $\deg(h)$ divise q^{n-1} (et h divise $Q_{q^{n-1}}$). On a donc

$$Q_{q^n} = Q_{q^{n-1}} \times \tilde{Q}_{q^n}.$$

Ainsi,

$$\deg(\tilde{Q}_{q^n}) = \deg(Q_{q^n}) - \deg(Q_{q^{n-1}}) = p^{q^n} - p^{q^{n-1}}.$$

Il y a donc $(p^{q^n} - p^{q^{n-1}})/q^n$ polynômes irréductibles unitaires de degré q^n parmi les p^{q^n} polynômes unitaires de $\mathbb{F}_p[x]$ de degré q^n, soit une probabilité d'irréductibilité de

$$\frac{p^{q^n} - p^{q^{n-1}}}{q^n p^{q^n}} = \frac{1}{q^n}\Big(1 - \frac{1}{p^{q^n - q^{n-1}}}\Big).$$

Par exemple, un polynôme de degré 4 dans $\mathbb{F}_2[x]$ a une probabilité 3/16 d'être irréductible.

Le cas général. Le cas général est plus compliqué. On a la formule

$$Q_d = \prod_{e|d} \tilde{Q}_e$$

et il faut en général connaître les degrés de tous les $\tilde{Q}_e$ pour e divisant d, $e < d$ pour en déduire le degré de $\tilde{Q}_d$ (donc le nombre de polynômes irréductibles de degré d). On peut cependant montrer facilement que la probabilité $P(d,p)$ qu'un polynôme unitaire de degré d dans $\mathbb{F}_p[x]$ soit irréductible satisfait les estimations

$$\frac{1}{2d} \leq \frac{1}{d}\Big(1 - \frac{2}{p^{d/2}}\Big) \leq P(n,p) \leq \frac{1}{d}$$

dès lors que $p^d \geq 16$ (GG, p.398).

Remarque : On remarque que la probabilité que $f \in \mathbb{F}_p[x]$ unitaire de degré d une puissance d'un nombre premier soit irréductible :

- croît avec p, et tend vers $\frac{1}{d}$ quand p tend vers l'infini (d'autant plus vite que d est grand).
- décroît avec q, et tend vers 0 quand q tend vers l'infini : les polynômes irréductibles se raréfient quand d augmente, tout comme les nombres premiers se raréfient dans la suite des nombres entiers. Il est donc naturel de se poser la question de comment fabriquer un polynôme irréductibles grand degré.

4.3 Fabriquer un polynôme irréductible

Il existe des algorithmes probabilistes (Ben-Or, etc) performants qui permettent de construire des polynômes irréductibles de degré d grand à coefficients dans $\mathbb{F}_p$. L'idée toute simple est de choisir un polynôme aléatoire unitaire de degré d dans $\mathbb{F}_p[x]$ puis de tester son irréductibilité, et ce jusqu'à obtenir un polynôme irréductible. Les meilleures complexités actuelles se situent autour de $\mathcal{O}(d^2 \log(p))$ opérations dans $\mathbb{F}_p$. Ils utilisent le fait que les facteurs de plus petits degrés sont en moyenne de degré $\mathcal{O}(\log(d))$.

5 Exercices et TP

Dans tout ce qui suit, on adopte les notations suivantes :

- $F \in \mathbb{Z}[x]$ désigne un polynôme quelconque à coefficients entiers.
- $p \in \mathbb{N}$ désigne un nombre premier.
- $f \in \mathbb{F}_p[x]$ désigne la classe de F modulo p.

Pour travailler modulo p avec Maxima, on pose **modulus** :p, puis f :**polymod**(F) (ou **f :polymod(F,p)** renvoie la réduction de F modulo p. Attention, il arrive que Maxima traite f comme un polynôme de $\mathbb{Z}[x]$. Par exemple **expand**(**polymod**($(x+3)^2, 2$)) renvoie x^2+2x+1 et non x^2+1. A vous de vous familiariser avec les fonctions **modulus** et **polymod**. On rappelle que l'on accède aux informations relatives à un mots clé, en tapant ce mot clé précédé de **?** ou **? ?**.

On aura besoin au préalable d'une procédure de puissance rapide dans l'anneau $R = \mathbb{F}_p[x]/(f)$.

Exercice 0 Implémenter un algorithme **Puissance**(G, n, F, p) qui, donnés $G \in \mathbb{Z}[x]$, $n \in \mathbb{N}$ et $F \in \mathbb{Z}[x]$ et $p \in \mathbb{N}$ premier, calcule la puissance rapide $\bar{g}^n$ de $\bar{g} := G \mod (p, F)$ dans l'anneau $R = \mathbb{F}_p[x]/(f)$. Définir g : **polymod**(G) et utiliser la fonction **remainder**(g, f) pour travailler modulo f.

5.1 La méthode de Berlekamp

On suppose ici que f est unitaire sans facteurs multiples.

Exercice 1 1) Ecrire un algorithme **Matrice**(F,p) qui calcule la matrice de l'endomorphisme

$$\phi_p - Id : R \to R, \qquad g \mapsto g^p - g$$

dans la base canonique $(1, x, x^2, \ldots, x^{d-1})$ de $R = \mathbb{F}_p[x]/(f)$ ($d = \deg(f)$ et $x = x \mod f$ par abus de notation). Utiliser la fonction **zeromatrix**(d, d) qui renvoie la matrice nulle de taille $d \times d$. Remplir cette matrice avec les coordonnées des vecteurs $(\phi_p - Id)(x^i)$ à l'aide de la fonction **coeff**(g, x, k) de Maxima qui renvoie le coefficient de x^k dans g.

2) En déduire un algorithme **Noyau**(F, p) qui renvoie la liste $[g_1, \ldots, g_s]$ des vecteurs d'une base de la sous-algèbre de Berlekamp

$$\mathcal{B} := \{g \in R,\ g^p - g = 0\}.$$

Utiliser la fonction **nullspace**(M) de Maxima qui renvoie une base du noyau de la matrice M. Extraire les vecteurs de la base à l'aide de la fonction **args** qui renvoie la base sous forme de liste, et transformer les vecteurs $v_i = args[i]$ en les polynômes g_i avec **sum**$(v_i[j] * x^j, j, 0, d-1)$.

Exercice 2 En s'appuyant sur la méthode de Berlekamp déterministe décrite dans le cours, et en utilisant la procédure **Noyau**, implémenter un algorithme **Berlekamp1**(F, p) qui retourne $[f_1, \ldots, f_s]$.

Exercice 3 En s'appuyant sur la méthode de Berlekamp probabiliste décrite dans le cours, et en utilisant la procédure **Noyau**, implémenter un algorithme **Berlekamp2**(F,p) qui retourne $[f_1, \ldots, f_s]$. Utiliser une sous-procédur'unee **Cherche**(F, p, L) qui renvoie la première factorisation non triviale $f = f_1 f_2$ obtenue à partir de la liste $L =$**Noyau**(F, p) en utilisant la fonction **Random** de Maxima et en itérant la méthode probabiliste décrite dans le cours.

5.2 La méthode de Cantor-Zassenhaus

On suppose ici $f = F \mod (p)$ unitaire sans facteurs carrés.

Exercice 4 Ecrire une procédure **DD**(F, p) (Distinct Degree factorization) qui renvoie la liste

$$DD(F, p) := [h_1, h_2, \ldots, h_n],$$

avec les $h_i \in \mathbb{F}_p[x]$ unitaires, produits des facteurs irréductibles de f de degrés i, $f = h_1 \cdots h_n$ et $h_n \neq 1$.

Exercice 5 Ecrire une procédure **EqualDeg**(F, p) basée sur la méthode de Cantor-Zassenhaus qui, donné $F \mod (p)$ produit de facteurs irréductibles f_i de même degrés, renvoie $[f_1, \ldots, f_s]$.

Exercice 6 En déduire une procédure **Zassenhaus**(F, p) qui retourne $[f_1, \ldots, f_s]$.

5.3 Traiter les facteurs multiples

Exercice 7 Pour avoir la possibilité d'utiliser l'un ou l'autre des 3 algorithmes de factorisation des polynômes sans facteurs multiples développés ci-dessus, écrire une procédure **Facto**$(F, p, choix)$ qui renvoie **Berlekamp1**$(F, p, choix)$ si $choix = 1$, **Berlekamp2**(F, p) si $choix = 2$, et **Zassenhaus**(F, p) si $choix = 3$.

On appelle **factorisation** de F modulo p la liste

$$L(F, p) = [[f_1, m_1], \ldots, [f_s, m_s]]$$

des paires consituées des facteurs irréductibles unitaires de f et de leurs multiplicités.

Exercice 8 Implémenter un algorithme **TraiteB**$(F, p, choix)$ basé sur **Facto**$(F, p, choix)$ qui retourne $[b, L(b, p)]$ où b est le facteur de f décrit dans le cours.

Exercice 9 Implémenter un algorithme **TraiteA**(A, p) qui prend en entrée p premier et $A \in \mathbb{Z}[x]$ tel que $A' = 0 \mod (p)$, et qui retourne l'unique paire (g, q) telle que $q \in \{p, p^2, \ldots\}$, $g' \neq 0$ et $A = g^q \mod (p)$.

5.4 L'algorithme global

Exercice 10 Implémenter un algorithme **FactoMonic**$(F, p, choix)$ basé sur les algorithmes précédants qui, donné $F \in \mathbb{Z}[x]$ *unitaire* retourne $L(F, p)$.

Exercice 11 1) En déduire un algorithme **FactoListe**$(F, p, choix)$ basé sur les algorithmes précédants qui, donnés $F \in \mathbb{Z}[x]$ et p premier, retourne $[c, L(F, p)]$.

2) Déduire un algorithme **FactoProd**$(F, p, choix)$ qui renvoie la factorisation de f sous forme de produit $cf_1^{m_1} \cdots f_s^{m_s}$.

5.5 Tests et comparaisons de complexité

On se propose de comparer les trois algorithmes de factorisation précédants avec l'algorithme **factor** de Maxima. On va pour cela calculer leurs temps moyens d'exécution sur des listes aléatoires de polynômes dépendant de divers paramètres.

Exercice 12 Définir une liste $FAC := [Facto1, Facto2, Facto3, Facto4]$, avec

$\mathbf{Facto1}(F,p) := \mathbf{FactoProd}(F,p,1)$,
$\mathbf{Facto2}(F,p) := \mathbf{FactoProd}(F,p,2)$,
$\mathbf{Facto3}(F,p) := \mathbf{FactoProd}(F,p,3)$,
$\mathbf{Facto4}(F,p) := \mathbf{factor}(\mathbf{polymod}(F,p))$.

Exercice 13 Ecrire une procédure $\mathbf{AleaListe}(m,k,n,p)$ qui renvoie une liste de m polynômes produits de k polynômes de degrés n à coefficients entiers aléatoires dans l'intervalle $[0, p-1]$.

Exercice 14 1) Ecrire une procédure $\mathbf{TempsTotal}(m,k,n,p)$ qui renvoie les temps **totaux** de chaque fonction $FAC[i]$ appliquée à f, $i = 1, \ldots, 4$ lorsque f parcourt $\mathbf{AleaListe}(m,n,k,p)$.

P.S : Pour récupérer le temps d'exécution d'une fonction G :

a) $\boldsymbol{timer}(G)$ *pour dire à Maxima de calculer les temps/statistiques de la fonction G.*

b) t *:*$\boldsymbol{get}(G, runtime)$ *qui donne le temps total d'exécution de la fonction G depuis qu'elle est dans* ***timer****. Attention, si on veut le temps d'une seule exécution de G, il faut retrancher le temps avant exécution au temps après exécution.*

2) En déduire une procédure $\mathbf{Complexité}(m,k,n,p)$ qui, pour chaque fonction $FAC[i]$, calcule la liste $[t(1), \ldots, t(n)]$ des temps **moyens** d'exécution lorsque F parcourt les n listes $\mathbf{AleaListe}(m,k,i,p)$, $i = 1, \ldots, n$. Placer les points $\{(ki, t(i)), i = 1, \ldots, n\}$ dans un même graphique pour chaque fonction.

P.S : Pour afficher les points $(i, t(i))$ *pour* $i = 1, \ldots, n$.

a) Créer la liste L : $[[1, t(1)], \ldots, [n, t(n)]$

b) Taper $wxplot2d([discrete, L])$.

c) Taper $wxplot2d([[discrete, L1], [discrete, L2], ...])$ *pour afficher les points de plusieurs listes sur un même graphique. Voir plot − options pour ajouter diverses options (couleurs, légende, etc.).*

3) Exécuter la fonction **Complexité** avec les valeurs (m,k,n,p) variant (par exemple) dans l'ensemble

$$\{(1,1,15,3), (20,1,15,3), (5,10,3,3), (5,10,3,103)\}$$

et comparer les performances respectives de chaque algorithme en fonctions des différents paramètres.

5.6 Applications : racines, irréductibilité,...

Exercice 15 Ecrire une procédure $\mathbf{Racines}(F,p)$ qui renvoie la liste des racines de f dans $\mathbb{F}_p$.

Exercice 16 Ecrire une procédure $\mathbf{Irreductible}(F,p)$ qui teste si f est irréductible ou réductible. Calculer la proportion de polynômes irréductibles sur $\mathbf{AleaListe}(1000,1,17,5)$ et comparer avec la probabilité calculée dans le cours.

Exercice 17 Ecrire une procédure **RandomIrred**(d, p) qui renvoie un polynôme unitaire irréductible aléatoire de $\mathbb{F}_p[x]$ de degré d.