2.13 Algorithme de Berlekamp (122, 123, 141, 142, 148) [3]

Dans ce développement, on détaille un algorithme pour factoriser un polynôme sans facteurs carrés à coefficients dans un corps fini. On verra également que la condition d'être sans facteurs carrés n'est pas limitante. Cet algorithme se base sur le théorème chinois et l'algèbre linéaire. Il rentre donc bien dans les leçons 122 et 148 selon moi ! On fixe alors, dans tout ce développement, un nombre premier p, un entier n et un corps fini $\mathbb{F}_q$ de cardinal $q = p^n$.

Lemme 2.32 (Élévation à la puissance q)**.** Soit $R \in \mathbb{F}_q[X]$ un polynôme. L'application :

$$\begin{array}{rccc} S_R & : & \dfrac{\mathbb{F}_q[X]}{(R)} & \longrightarrow & \dfrac{\mathbb{F}_q[X]}{(R)} \\ & & \overline{Q}^R & \longmapsto & \overline{Q\left(X^q\right)}^R \end{array}$$

est bien définie, linéaire et on a :

$$\forall Q \in \mathbb{F}_q[X], \quad S_R\left(\overline{Q}^R\right) = \overline{Q^q}^R.$$

Démonstration. Par la propriété universelle des anneaux de polynômes, le morphisme de $\mathbb{F}_q$-algèbres suivant :

$$\begin{array}{rccc} S & : & \mathbb{F}_q[X] & \longrightarrow & \mathbb{F}_q[X] \\ & & a \in \mathbb{F}_q & \longmapsto & a \\ & & X & \longmapsto & X^q \end{array}$$

est bien défini et les deux conditions données ci-dessus le caractérisent. Il s'agit l'application $Q \mapsto Q(X^q)$. Cependant, l'application :

$$\begin{array}{ccc} \mathbb{F}_q[X] & \longrightarrow & \mathbb{F}_q[X] \\ Q & \longmapsto & Q^q \end{array}$$

est également un morphisme de $\mathbb{F}_q$-algèbres vérifiant ces deux conditions. En effet, la compatibilité par produit est claire et étant donné que pour tout $a \in \mathbb{F}_q$, $a^q = a$ par le théorème de Lagrange, on a :

$$\forall a \in \mathbb{F}_q,\ \forall Q \in \mathbb{F}_q[X], \quad (aQ)^q = aQ^q.$$

Enfin, la compatibilité par somme se démontre ainsi :

$$\forall Q, S \in \mathbb{F}_q[X], \quad (Q+S)^q = \varphi^n(Q+S)$$

où φ est le morphisme de Frobenius étendu aux polynômes :

$$\forall Q \in \mathbb{F}_q[X], \quad \varphi(Q) = Q^p.$$

Ainsi, étant donné que :

$$\varphi(Q+S) = \sum_{i=0}^{p} \binom{p}{i} Q^i S^{p-i}$$

et que :

$$\forall i \in [\![1, p-1]\!], \quad \binom{p}{i} \equiv 0\ [p],$$

on a :

$$\varphi(Q+S) = \varphi(Q) + \varphi(S).$$

On obtient donc :

$$\forall Q, S \in \mathbb{F}_q[X], \quad (Q+S)^q = \varphi^n(Q+S) = \varphi^n(Q) + \varphi^n(S) = Q^q + S^q.$$

On a donc que les applications $Q \mapsto Q\left(X^q\right)$ et $Q \mapsto Q^q$ sont égales. Ainsi, en considérant la surjection canonique :

$$\pi_R \quad : \quad \mathbb{F}_q[X] \quad \longrightarrow \quad \frac{\mathbb{F}_q[X]}{(R)}$$

on a que l'application $\pi_R \circ S$ est un morphisme de $\mathbb{F}_q$-algèbres dont le noyau contient (R). En effet :

$$S(R) = R^q \equiv 0 \ [R].$$

$\pi_R \circ S$ passe donc au quotient en un endomorphisme S_R de $\mathbb{F}_q$-algèbres de $\dfrac{\mathbb{F}_q[X]}{(R)}$, qui coïncide avec l'élévation à la puissance q, ce qui conclut. □

On est prêt à prouver les résultats fondamentaux assurant la correction et la terminaison de l'algorithme de Berlekamp.

Théorème 2.33. Soit $P \in \mathbb{F}_q[X]$ un polynôme sans facteur carré. Écrivons-le comme produit de ses facteurs irréductibles :

$$P = \prod_{i=1}^{r} P_i.$$

Alors :

1. $r = \dim\left(\ker\left(S_P - id_{\frac{\mathbb{F}_q[X]}{(P)}}\right)\right) = \deg(P) - \operatorname{rg}\left(S_P - id_{\frac{\mathbb{F}_q[X]}{(P)}}\right)$.
2. Si $r > 1$, alors il existe $V \in \mathbb{F}_q[X]$ tel que V ne soit pas congru modulo P à un polynôme constant et tel que :
$$\overline{V}^P \in \ker\left(S_P - id_{\frac{\mathbb{F}_q[X]}{(P)}}\right).$$
Il existe alors un $\alpha \in \mathbb{F}_q$ tel que $P \wedge (V - \alpha)$ soit un diviseur non trivial de P.

Démonstration. 1. On considère, pour $i \in [\![1, r]\!]$ le quotient :

$$K_i := \frac{\mathbb{F}_q[X]}{(P_i)}.$$

Étant donné que le polynôme P_i est irréductible sur le corps $\mathbb{F}_q$, l'idéal (P_i) est maximal et donc le quotient K_i est en fait un corps. De plus, étant donné que les P_i sont deux à deux premiers entre eux, le lemme chinois s'applique et on a à disposition un isomorphisme de $\mathbb{F}_q$-algèbres :

$$\begin{array}{rccc} \Psi & : & \dfrac{\mathbb{F}_q[X]}{(P)} & \longrightarrow & \displaystyle\prod_{i=1}^{r} K_i \\ & & \overline{Q}^P & \longmapsto & \left(\overline{Q}^{P_1}, \ldots, \overline{Q}^{P_r}\right). \end{array}$$

Le fait que les K_i soient des corps est intéressants car on peut les considérer comme des extensions du corps $\mathbb{F}_q$, dans lesquelles les éléments de $\mathbb{F}_q$ s'identifient exactement aux éléments de K_i invariants par élévation à la puissance q. En effet, dans un corps, le polynôme $X^q - X$ a au plus q racines, et les éléments de $\mathbb{F}_q$ sont exactement les racines de ce polynôme. Tiens, tiens ! À quoi ressemble l'application $\tilde{S}_P := \Psi \circ S_P \circ \Psi^{-1}$ au

fait ?

$$\forall Q \in \mathbb{F}_q[X], \quad \tilde{S}_P\left(\overline{Q}^{P_1}, \dots, \overline{Q}^{P_r}\right) = \left(\overline{Q^q}^{P_1}, \dots, \overline{Q^q}^{P_r}\right) = \left(\left(\overline{Q}^{P_1}\right)^q, \dots, \left(\overline{Q}^{P_r}\right)^q\right)$$

Il s'agit donc de l'élévation à la puissance q dans le produit $\prod_{i=1}^{r} K_i$! Ah mais on voulait pas calculer la dimension des points fixes de l'élévation à la puissance q nous ? Eh mais ce serait pas juste $\mathbb{F}_q^r$ du coup ? ? Presque ! Détaillons-le proprement. Puisque Ψ est un isomorphisme, on a :

$$\ker\left(\tilde{S}_P - id_{\prod_{i=1}^r K_i}\right) = \Psi\left(\ker\left(S_P - id_{\frac{\mathbb{F}_q[X]}{(P)}}\right)\right)$$

et donc les dimensions sont conservées. Ainsi :

$$\begin{aligned}
\dim\left(\ker\left(S_P - id_{\frac{\mathbb{F}_q[X]}{(P)}}\right)\right) &= \dim\left(\ker\left(\tilde{S}_P - id_{\prod_{i=1}^r K_i}\right)\right) \\
&= \dim\left(\left\{(x_1, \dots, x_r) \in \prod_{i=1}^{r} K_i \ \mid \forall i \in [\![1, r]\!],\ x_i^q = x_i\right\}\right)
\end{aligned}$$

Or, d'après notre remarque, les éléments de K_i fixes par élévation à la puissance q sont exactement les éléments de la "copie" de $\mathbb{F}_q$ dans K_i. Cette "copie" est isomorphe en tant que $\mathbb{F}_q$-espace vectoriel à $\mathbb{F}_q$, ce qui nous suffit pour dire :

$$\begin{aligned}
\dim\left(\ker\left(S_P - id_{\frac{\mathbb{F}_q[X]}{(P)}}\right)\right) &= \dim\left(\left\{(x_1, \dots, x_r) \in \prod_{i=1}^{r} K_i \ \mid \forall i \in [\![1, r]\!],\ x_i^q = x_i\right\}\right) \\
&= \dim\left(\mathbb{F}_q^r\right) \\
&= r\,!
\end{aligned}$$

cela conclut la preuve du premier résultat.

2. Si $r > 1$, alors $\ker\left(S_P - id_{\frac{\mathbb{F}_q[X]}{(P)}}\right)$ n'est pas réduit à une droite vectorielle. Ainsi, ce sous-espace vectoriel contient strictement la droite :

$$\text{Vect}\left(\overline{1}^P\right) = \left\{\overline{a}^P \ \mid a \in \mathbb{F}_q\right\}$$

des résidus modulo P des polynômes congrus à une constante modulo P. Ainsi, il existe un polynôme $V \in \mathbb{F}_q[X]$ non congru à une constante modulo P tel que $\overline{V}^P \in \ker\left(S_P - id_{\frac{\mathbb{F}_q[X]}{(P)}}\right)$. En repassant dans l'anneau produit $\prod_{i=1}^{r} K_i$, on a donc que V n'est pas congru à une constante modulo P, mais que les résidus $\overline{V}^{P_i}$ sont fixes par élévation à la puissance q, ce sont donc des éléments de la copie de $\mathbb{F}_q$ dans K_i ! Autrement dit :

$$\forall i \in [\![1, r]\!],\ \exists \alpha_i \in \mathbb{F}_q, \quad V \equiv \alpha_i\ [P_i].$$

De plus, cet α_i est unique (c'est le reste de la division euclidienne de V par P_i). Cela signifie donc que P_i divise $V - \alpha_i$ et donc P_i apparaît dans la décomposition en facteurs irréductibles de $P \wedge (V - \alpha_i)$! De plus, comme V n'est pas congru à une constant modulo P, on a que P ne divise pas $V - \alpha_i$ et donc les polynômes

$$P \wedge (V - \alpha_i)$$

sont tous des diviseurs non triviaux de P ! Cela conclut donc la preuve du théorème.

□

Montrons alors que l'algorithme de Berlekamp :

```
def Berlekamp(P):
```

```
if pgcd(P,deriv(P)) != 1:
    return('Ce polynôme a un facteur carré !')
else:
    n = deg(P)
    r = dim(ker(SP-np.eye(n)))
    if r == 1:
        return [P]
    else:
        V = LeBonPolynome(P) #Calcule un polynôme non constant modulo P comme au point 2.
        for alpha in Fq:
            B = pgcd(P,V-alpha)
            if B != 1:
                Q = P/B
                return (Berlekamp(Q) + Berlekamp(B))
                break
```

termine et est correct. Lorsqu'on doit appeler l'algorithme de Berlekamp dans notre boucle conditionnelle, on l'applique aux polynômes $Q = \frac{P}{P \wedge (V-\alpha)}$ et $B := P \wedge (V - \alpha)$ qui possèdent tous deux strictement moins de facteurs irréductibles que P. En effet, d'après ce qu'on a vu au point 2., on a :

$$P \wedge (V - \alpha) = \prod_{i \in I_\alpha} P_i$$

où $I_\alpha \subset [\![1, r]\!]$ est un sous-ensemble non-vide et strictement inclus dans $[\![1, r]\!]$ car $P \wedge (V - \alpha) \neq P$. Ainsi, Q possède $r - |I_\alpha| < r$ facteurs irréductibles et B en possède $|I_\alpha| < r$. Ainsi, l'algorithme termine. Montrons donc que l'algorithme est correct, par récurrence forte sur r. L'initialisation est claire : si $r = 1$, P est irréductible et l'algorithme renvoie la liste $[P]$, ce qui est correct. Si $r > 1$, on a la décomposition :

$$P = QB = \prod_{i \in [\![1,r]\!] \setminus I_\alpha} P_i \times \prod_{i \in I_\alpha} P_i.$$

Ainsi, étant donné que les P_i sont irréductibles distincts, en notant $\mathrm{FI}(P)$ l'ensemble des facteurs irréductibles (à association près) de P, on a la partition :

$$\mathrm{FI}(P) = \mathrm{FI}(Q) \sqcup \mathrm{FI}(B),$$

ce qui montre que la ligne 15 est correcte. Étant donné que Q et B possèdent strictement moins de facteurs irréductibles que P, l'hypothèse de récurrence montre que l'algorithme de Berlekamp renvoie bien les facteurs irréductibles de Q et B. Ainsi, l'algorithme renvoie bien les facteurs irréductibles de P, ce qui conclut !

Remarque 2.13.1 (Quelle est la complexité de l'algorithme de Berlekamp ?)**.** *Pour faire marcher l'algorithme, on doit calculer un noyau et plusieurs PGCD. On doit alors calculer la matrice de l'application $S_P - id$ dans la base :*

$$\mathscr{B} = \left(\overline{1}^P, \overline{X}^P, \ldots, \overline{X^{\deg(P)-1}}^P\right)$$

et effectuer un piovt de Gauss sur cette matrice, ce qui nous fait $O\left(\deg(P)^3\right)$ opérations. Il faut ensuite boucler sur les éléments de $\mathbb{F}_q$ et calculer un PGCD. L'algorithme d'Euclide pour deux polynômes de degré au plus N à coefficients dans un corps se fait en $O(N^2)$ opérations (j'ai trouvé ça dans un poly de l'université de Bordeaux). Ainsi, un appel de l'algorithme de Berlekamp utilise $O\left(\deg(P)^3 + q\deg(P)^2\right)$ opérations.

Comment factoriser un polynôme quelconque et pas juste un polynôme sans facteur carré ? On peut trouver un algorithme utilisant l'algorithme de Berlekamp pour factoriser n'importe quel polynôme sur $\mathbb{F}_q[X]$. En effet, P est sans facteur carré si et seulement si $P \wedge P' = 1$ (cf. développement sur la décomposition de Dunford/Jordan-Chevalley) et dans ce cas, on peut appliquer l'algorithme de Berlekamp. Si $P \wedge P' \neq 1$, alors :

— ou bien $P \wedge P' \neq P$, dans ce cas on peut appliquer notre algorithme de factorisation aux polynômes $P \wedge P'$ et $\frac{P}{P \wedge P'}$,
— ou bien $P \wedge P' = P$, ce qui signifie que $P' = 0$. On a alors le résultat suivant :

Lemme 2.34 (Polynôme dérivé nul)**.** Soit $P \in \mathbb{F}_q[X]$ avec $q = p^n$, p étant un nombre premier. Les deux conditions suivantes sont équivalentes :

1. $P' = 0$,
2. $\exists R \in \mathbb{F}_q[X], \quad P = R^p$.

Démonstration. 2. $\Rightarrow$ 1. est clair : $P' = pR'R^{p-1} = 0$
1. $\Rightarrow$ 2. Posons :

$$P = \sum_{i=0}^{n} a_i X^i.$$

la condition $P' = 0$ donne :

$$P' = \sum_{i=1}^{n} i a_i X^{i-1} = 0$$

i.e.

$$\forall i \in [\![0, n]\!], \quad p \nmid i \Longrightarrow a_i = 0.$$

Ainsi :

$$P = \sum_{j=0}^{\lfloor \frac{n}{p} \rfloor} a_{pj} X^{pj}.$$

Or, dans un corps fini, le Frobenius (élévation à la puissance p) est bijectif. Si φ désigne le Frobenius, son inverse est simplement φ^{n-1}. Ainsi, en notant $m := \left\lfloor \frac{n}{p} \right\rfloor$:

$$P = \sum_{j=0}^{m} a_{jp} X^{jp} = \left(\sum_{j=0}^{m} \varphi^{n-1}(a_{jp}) X^j \right)^p = R^p$$

où

$$R = \sum_{j=0}^{m} \varphi^{n-1}(a_{jp}) X^j$$

□

On a alors l'algorithme de factorisation suivant :

```
def Factorisation(P):
    Q = pgcd(P,deriv(P))
    if Q == 1:
        return Berlekamp(P)
    elif Q != P:
        return Factorisation(Q) + Factorisation(P/Q)
    else:
        return Factorisation(P**(1/p)).
```