Agrégation externe de mathématiques, session 2019
Épreuve de modélisation, option C : Algèbre et Calcul Formel

(*C03-public*)

Résumé : On étudie une forme particulière de transformée de Fourier, adaptée à la multiplication de polynômes en caractéristique 2. Les sous-groupes des unités sont remplacés par des sous-groupes additifs du corps de base.

Mots clefs : corps finis, arithmétique des polynômes

► *Il est rappelé que le jury n'exige pas une compréhension exhaustive du texte. La présentation, bien que totalement libre, doit être organisée et le jury apprécie qu'un plan soit annoncé en préliminaire. L'exposé doit être construit en évitant la paraphrase et mettant en lumière les connaissances, à partir des éléments du texte. Il doit contenir des illustrations informatiques réalisées sur ordinateur, ou, à défaut, des propositions de telles illustrations. Des pistes de réflexion, indicatives et largement indépendantes les unes des autres, vous sont proposées en fin de texte.*

1. Introduction

La multiplication est une opération cruciale en algorithmique. En effet, la plupart des opérations arithmétiques s'y ramènent. Par exemple, la vitesse d'exécution de la division ou de la racine carrée effectuée par une itération de Newton est proportionnelle à celle de la multiplication (et de l'addition, opération *a priori* plus facile).

Lorsque l'on multiplie de très gros objets, que ce soient des entiers ou des polynômes, les méthodes s'appuyant sur la transformée de Fourier sont les plus efficaces et sont implantées dans la plupart des logiciels de calcul formel modernes. Le temps de calcul pour une multiplication est alors quasiment proportionnel à la taille des objets à multiplier : pour multiplier deux entiers de d chiffres ou deux polynômes de degré d, le temps de calcul croît quasi-linéairement avec d, en $O(d(\log d)^C)$, pour un certain C.

En théorie des codes correcteurs d'erreurs et en cryptographie, on manipule souvent des objets en caractéristique 2. Ceux-ci ne s'injectent pas naturellement dans le corps des complexes où est usuellement définie la transformée de Fourier. De fait, les algorithmes de transformée de Fourier rapide s'appuyent en général sur des racines 2^k-ème de l'unité, ce qui pose problème en caractéristique 2.

On s'intéresse ici à un algorithme de multiplication en caractéristique 2, inspiré des méthodes à base de transformée de Fourier, mais où le groupe multiplicatif, qui contient les racines de l'unité utilisées dans l'algorithme classique, est remplacé par le groupe additif.

2. Sous-groupes additifs de $\mathbb{F}_{2^n}$

La notion clef pour la suite est celle de polynôme linéarisé :

Définition 1. *Soit K un corps de caractéristique p. Un* polynôme linéarisé *sur K est un polynôme non nul de $K[x]$ dont tous les monômes sont de degré une puissance de p.*

Soit $\mathbb{F}_{2^n}$ le corps fini à 2^n éléments. Soit W un sous-groupe de $(\mathbb{F}_{2^n}, +)$. Son cardinal est de la forme 2^m. On note $w(x)$ l'unique polynôme unitaire de degré 2^m dont les racines sont exactement les éléments de W. Le théorème suivant permet d'affirmer qu'il s'agit d'un polynôme linéarisé.

Théorème 1. *Soit $w(x)$ un polynôme scindé, sans racines multiples, sur $\mathbb{F}_{2^n}$. Alors $w(x)$ est linéarisé si et seulement si ses racines forment un sous-groupe additif de $\mathbb{F}_{2^n}$.*

Démonstration. Le fait que les racines d'un polynôme linéarisé forment un groupe découle simplement de la linéarité de $x \mapsto x^2$. Pour la réciproque, on considère le déterminant suivant :

$$D(x) = \begin{vmatrix} b_1 & b_1^2 & \cdots & b_1^{2^m} \\ \vdots & \vdots & & \vdots \\ b_m & b_m^2 & \cdots & b_m^{2^m} \\ x & x^2 & \cdots & x^{2^m} \end{vmatrix},$$

où les b_i engendrent le sous-groupe. On montre par récurrence sur m que le coefficient dominant de ce polynôme n'est pas nul donc que le polynôme n'est pas identiquement nul. De plus, l'ensemble de ses racines est le groupe engendré par les b_i. □

Un sous-groupe additif W d'ordre 2^m de $\mathbb{F}_{2^n}$ peut être muni d'une structure d'espace vectoriel sur $\mathbb{F}_2$ de dimension m. Soit $(b_1, \ldots, b_m)$ une base de W ; on lui associe la suite de sous-espaces vectoriels définie par $W_0 = \{0\}$, et $W_i = W_{i-1} \oplus \langle b_i \rangle$ pour $i \in [1, m]$, de sorte que

$$\{0\} = W_0 \subset W_1 \subset \cdots \subset W_m = W.$$

On appelle *drapeau* une telle suite, associée à une base $(b_1, \ldots, b_m)$ de W.

On associe à chacun des sous-groupes W_i le polynôme linéarisé $w_i(x)$ qui lui correspond par le Théorème 1. La chaîne d'inclusions devient une chaîne de divisibilité de polynômes :

$$x = w_0(x) \mid w_1(x) \mid \cdots \mid w_m(x).$$

Le drapeau canonique.

Nous allons choisir une base $(b_1, \ldots, b_m)$ particulièrement adaptée aux calculs. On la définit indirectement, en construisant les polynômes associés au drapeau correspondant.

Soit $(s_i(x))_{i \geqslant 0}$ la suite de polynômes de $\mathbb{F}_2[x]$ définis par

$$s_0(x) = x \quad \text{et} \quad \forall\, i \geqslant 0,\ s_{i+1}(x) = s_i(x)^2 + s_i(x).$$

Proposition 2. *Pour tout i, le polynôme s_i est sans facteur carré. De plus, pour tout ν,*

$$s_{2^\nu}(x) = x^{2^{2^\nu}} + x.$$

Démonstration. Par récurrence, en utilisant le fait que $s_{i+j}(x) = s_i \circ s_j(x)$. □

Dorénavant, le corps fini considéré est $\mathbb{F}_{2^n}$, où n est lui-même une puissance de 2 : $n = 2^v$. Ce corps est exactement formé des racines de $x^{2^n} + x = s_{2^v}(x)$. On associe à un tel corps, dont le degré est une puissance de 2, un *drapeau canonique*, formé des sous-groupes W_i associés aux polynômes $s_i(x)$. Bien que les W_i contiennent des éléments qui ne sont pas dans $\mathbb{F}_2$, les polynômes associés sont, eux, dans $\mathbb{F}_2[x]$.

3. Évaluation et interpolation rapide sur le drapeau canonique

Soient $P(x)$ et $Q(x)$ deux polynômes sur $\mathbb{F}_{2^n}$ que l'on souhaite multiplier. Supposons que le degré de leur produit est inférieur à $2^i \leqslant 2^n$. On procède par évaluation/interpolation :

(1) Évaluer : on calcule $(P(\alpha))_{\alpha \in W_i}$ et $(Q(\alpha))_{\alpha \in W_i}$;

(2) Multiplier point à point : calculer $((PQ)(\alpha))_{\alpha \in W_i}$;

(3) Interpoler : en déduire PQ.

L'étape (2) est très simple : on effectue 2^i produits dans $\mathbb{F}_{2^n}$. L'évaluation et l'interpolation, faites naïvement, nécessitent une quantité d'opérations $(+, \times, /)$ dans $\mathbb{F}_{2^n}$ qui est quadratique en le degré du résultat ; ce qui ne permettra pas de gagner par rapport à l'algorithme classique de multiplication de P et Q. On veut donc gagner sur ces étapes.

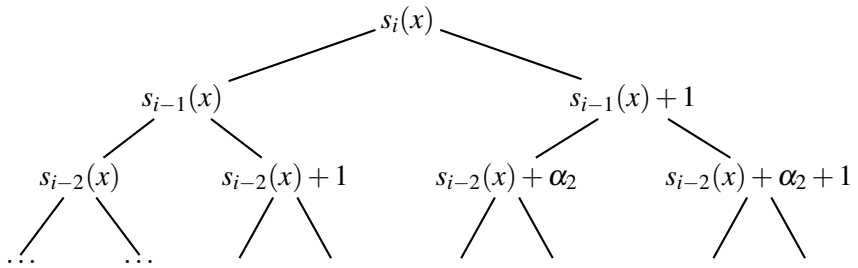
Évaluation. Supposons que l'on connaisse la réduction $P_{j,\alpha}(x)$ de $P(x)$ modulo $s_j(x)+\alpha$ pour un certain $j \leqslant i$ et $\alpha \in W_i$.

Alors, comme $s_j(x) = s_{j-1}(x)\left(s_{j-1}(x)+1\right)$, on a

$$s_j(x) + \alpha = \left(s_{j-1}(x) + \alpha'\right)\left(s_{j-1}(x) + \alpha' + 1\right),$$

avec $\alpha'^2 + \alpha' = \alpha$. On peut alors réduire $P_{j,\alpha}(x)$ modulo ces deux facteurs, afin d'obtenir $P_{j-1,\alpha'}$ et $P_{j-1,\alpha'+1}$. Cette étape se fait en un nombre d'opérations dans $\mathbb{F}_{2^n}$ qui est proportionnel au degré de $s_j(x)$, qui vaut 2^j, et au nombre de coefficients non nuls dans $s_{j-1}(x)$, qui est de l'ordre de j.

On peut ainsi faire «descendre» $P(x)$ dans l'arbre suivant, dont les feuilles sont les $s_0(x)+\alpha$ pour $\alpha \in W_i$.



À l'issue du calcul, on récupère aux feuilles les valeurs de $P(x)$ en tous les points de W_i. L'arbre a i niveaux ; pour passer du niveau $j-1$ au niveau j (qui contient des polynômes s_{i-j}), on doit effectuer de l'ordre de 2^j réductions, chacune coûtant de l'ordre de $2^{i-j}(i-j)$ opérations dans $\mathbb{F}_{2^n}$. En sommant, on obtient $O(2^i i^2)$ opérations dans $\mathbb{F}_{2^n}$.

Ainsi, l'évaluation de $P(x)$ en tous les points de W_i requiert $O(n^2 2^n)$ opérations, ce qui est bien mieux que l'algorithme naïf en $O(2^{2n})$.

Interpolation. Elle se traite de manière similaire, en remontant le long de l'arbre grâce à un théorème Chinois explicite à chaque étape.

En conclusion, on obtient le résultat suivant.

Théorème 2. *Si n est une puissance de 2, on peut multiplier deux polynômes de degré $d < 2^{n-1}$ sur $\mathbb{F}_{2^n}$ en $O(d(\log d)^2)$ opérations dans $\mathbb{F}_{2^n}$.*

4. Sur le nombre de coefficients binomiaux impairs

Dans l'algorithme ci-dessus, le temps de calcul est directement lié au nombre de coefficients non nuls dans les polynômes $s_i(x)$ décrivant le drapeau canonique.

Quelques expériences numériques suggèrent que la borne i utilisée dans les estimations du paragraphe précédent est pessimiste. On peut de fait améliorer celle-ci.

Remarquons d'abord que $s_i(x) = \sum_{j=0}^{i} \binom{i}{j} x^{2^j}$, où $\binom{i}{j}$ désigne le coefficient binomial réduit modulo 2. On est donc ramené à borner le nombre de coefficients binomiaux impairs.

Théorème 3. *L'exacte puissance de 2 qui divise le coefficient binomial $\binom{a+b}{a}$ est égale au nombre de retenues que l'on doit prendre en compte lorsque l'on additionne a et b, écrits en base 2.*

Démonstration. L'exacte puissance m_a de 2 qui divise $a!$ est

$$m_a = \sum_{i \geqslant 1} \left\lfloor \frac{a}{2^i} \right\rfloor .$$

Notons $a = a_0 + 2a_1 + 4a_2 + \cdots + 2^k a_k$ l'écriture binaire de a, où $a_i \in \{0,1\}$ pour tout $0 \leqslant i \leqslant k$, et $S_a = a_0 + a_1 + \cdots + a_k$ la somme des chiffres binaires de a. De l'écriture précédente de m_a, on déduit $m_a = a - S_a$.

Notons $b = b_0 + 2b_1 + 4b_2 + \cdots + 2^k b_k$ l'écriture binaire de b. Le processus d'addition de a et b en base 2 se décrit par les suites (c_i) et (ε_i) d'éléments de $\{0,1\}$ telles que $a_0 + b_0 = c_0 + 2\varepsilon_0$, puis pour tout $i \in [1,k]$,

$$\varepsilon_{i-1} + a_i + b_i = c_i + 2\varepsilon_i .$$

Ainsi, $a+b = c_0 + 2c_1 + 4c_2 + \cdots + c_k 2^k + \varepsilon_k 2^{k+1}$ et $S_a + S_b = S_{a+b} + (\varepsilon_0 + \cdots + \varepsilon_k)$. La puissance de 2 qui divise exactement $\binom{a+b}{a}$ est $m_{a+b} - m_a - m_b$ d'où le résultat. □

Il suit que le nombre de coefficients non nuls de $s_j(x)$ est 2^{S_j}, puis que le nombre total de coefficients non nuls dans l'ensemble des polynômes $s_j(x)$ pour $j \leqslant i$ est de l'ordre de $i^{\log 3/\log 2}$:

Théorème 4. *Si n est une puissance de 2, on peut multiplier deux polynômes de degré $d < 2^{n-1}$ sur $\mathbb{F}_{2^n}$ en $O(d(\log d)^{1.585})$ opérations dans $\mathbb{F}_{2^n}$.*

Suggestions et pistes de réflexion

► *Les pistes de réflexion suivantes ne sont qu'indicatives et il n'est pas obligatoire de les suivre. Vous pouvez choisir d'étudier, ou non, certains des points proposés, de façon plus ou moins approfondie, mais aussi toute autre question à votre initiative. Vos investigations comporteront une partie traitée sur ordinateur et, si possible, des représentations graphiques de vos résultats. À défaut, si vos illustrations informatiques n'ont pas abouti, il est conseillé d'expliquer ce que vous auriez souhaité mettre en œuvre.*

— Ne pas perdre trop de temps sur un exposé de la FFT complexe classique ; ce n'est pas le sujet du texte.
— Discuter l'affirmation de l'introduction sur les racines 2^k-ème de l'unité en caractéristique 2 : montrer qu'il n'en existe qu'une seule dans une clôture algébrique de $\mathbb{F}_2$. Plus généralement, que dire des racines p^k-ème de l'unité en caractéristique p ?
— Calculer les premiers polynômes définissant le drapeau canonique.
— Estimer expérimentalement la complexité de la multiplication dans $\mathbb{F}_2[x]$ (voire dans $\mathbb{F}_{2^n}[x]$) implantée dans le logiciel de calcul formel de votre choix ;
— Illustrer numériquement l'estimation du nombre de coefficients binomiaux impairs.
— Détailler les preuves des divers Théorèmes et de la Proposition.
— Compléter la description de l'algorithme en étudiant la phase d'interpolation.
— Étendre le Théorème 3 pour évaluer l'exacte puissance d'un nombre premier p qui divise un binomial, en considérant les retenues dans l'addition en base p.