

Algorithmes rapides pour deux nombres algébriques

Alin Bostan

GAGE, École polytechnique

travail en commun avec

P. Flajolet, B. Salvy et É. Schost

Les têtes d'affiche

- k corps, f et g dans $k[T]$.
- *somme composée* :

$$f \oplus g := \prod_{\substack{f(\alpha)=0 \\ g(\beta)=0}} (T - (\alpha + \beta)).$$

- *produit composé* :

$$f \otimes g := \prod_{\substack{f(\alpha)=0 \\ g(\beta)=0}} (T - \alpha\beta).$$

- *produit diamant* par $H \in k[X, Y]$:

$$f \diamond_H g := \prod_{\substack{f(\alpha)=0 \\ g(\beta)=0}} (T - H(\alpha, \beta)).$$

- ▶ polynômes de degré $D := \deg(f) \deg(g)$.
- ▶ coefficients dans k .

Objectif et motivations

Objectif :

Accélérer le calcul de $f \oplus g$, $f \otimes g$ et $f \diamond_H g$.

Motivations :

- théorie algébrique des nombres.
- construction de polynômes irréductibles.
- sommation symbolique.
- théorie de Galois effective.
- suites récurrentes linéaires.
- décalage de polynômes.
- polynômes de Graeffe.

Petit historique

- ▶ Collins (1983).
- ▶ Dvornicich & Traverso (1986).
- ▶ Brawley & Carlitz (1987).
- ▶ Brawley, Gao & Mills (1999).

Calcul classique

► résultants bivariés ou trivariés.

$$F1. \quad (f \oplus g)(x) = \text{Res}_y(f(x - y), g(y)).$$

$$F2. \quad (f \otimes g)(x) = \text{Res}_y(y^{\deg(g)} f(x/y), g(y)).$$

► Complexité : $O_{\log}(M(D)\sqrt{D})$.

$$F3. \quad (f \diamond_H g)(x) = \text{Res}_y \left(\text{Res}_z(x - H(y, z), f(z)), g(y) \right).$$

► Complexité : $O_{\log}(M(D)D)$.

- Ici $M(D)$ = complexité du produit de deux séries à précision D .
- O_{\log} = termes logarithmiques cachés.

Nouveaux résultats

Théorème [BoFlSaSc02]

Soit f et g polynômes unitaires dans $k[T]$ et soit $D = \deg(f) \deg(g)$.

Soit $H \in k[X, Y]$.

$\text{char}(k)$	$f \otimes g$	$f \oplus g$	$f \diamond_H g$
nulle ou $> D$	$O(M(D))$	$O(M(D))$	$D^2 + O(\sqrt{D}M(D))$
arbitraire \star	$O(M(D) \log(D))$	$\overset{\circ}{\circ}$	$2D^2 + O(\sqrt{D}M(D))$

Le symbole \star indique l'hypothèse supplémentaire:

“ $\text{char}(k) \geq$ multiplicités des racines du polynôme calculé”.

Une application

Contexte : la construction de polynômes irréductibles de grand degré sur un corps fini.

Lemme *Sur un corps fini, $f \oplus g$ est irréductible si et seulement si f et g sont irréductibles, de degrés premiers entre eux.*

- Algorithme de Shoup :
 1. construire des irréductibles de degrés puissances de nombres premiers.
 2. recombinaison ces polynômes.

- ▶ On diminue le coût de la deuxième partie :
de $O(D^{(\omega+1)/2})$ à $O_{\log}(D)$.

Idée de base

- Représenter h par ses *sommes de Newton*

$$N_s(h) := \sum_{h(\gamma)=0} \gamma^s.$$

- Dvornicich & Traverso : aller-retour coefficients – sommes de Newton, basé sur :

Lemme [Formules de Newton]

Si $h = T^D + h_1 T^{D-1} + \dots + h_D$, alors

$$N_s(h) = - \sum_{i=1}^{s-1} h_i N_{s-i}(h) - s h_s.$$

- Complexité : quadratique en D .
 - Marche en caractéristique nulle ou $> D$.
-
- Faire des conversions rapides !
 - Traiter le cas de la caractéristique petite !

Conversion rapide polynôme - sommés de Newton

- Pour $P \in k[T]$, on note $\text{rec}(P)$ son *polynôme réciproque* $T^{\deg(P)} P\left(\frac{1}{T}\right)$.

Lemme *La série génératrice*

$$\text{Newton}(h) := \sum_{s \geq 0} N_s(h) T^s$$

est rationnelle et égale à $\frac{\text{rec}(h')}{\text{rec}(h)} \in k[[T]]$.

Corollaire

Les $O(D)$ premières sommes de Newton de h peuvent être calculées en $O(M(D))$.

Conversion rapide sommes de Newton - polynôme, $\text{char}(k) = 0$

Exponentielle d'une série $F \in k[[T]]$:

- $\exp(F) := \sum_{s \geq 0} \frac{F^s}{s!}$, si $\text{char}(k) = 0$.
- $\exp(F) := \sum_{s \geq 0} \frac{F^s}{s!}$, si $p = \text{char}(k) \geq D$.

Lemme

Soit $h \in k[T]$, unitaire, de degré D , sur un corps de caractéristique nulle ou $> D$. Alors :

$$\text{rec}(h) = \exp \left(\int \frac{1}{T} \cdot \left(D - \text{Newton}(h) \right) \right).$$

Corollaire

Le polynôme h peut être calculé à partir de ses D premières sommes de Newton en $O(M(D))$.

Conversion rapide sommes de Newton - polynôme, $\text{char}(k) > 0$

Lemme

1. Si $h \in k[T]$, alors la suite $(N_s(h))_{s \geq 0}$ de ses sommes de Newton est récurrente linéaire ;
2. le polynôme minimal de cette suite est égal à

$$\frac{h}{\text{pgcd}(h, h')}.$$

Corollaire Soit $h \in k[T]$ unitaire, de degré D . Si h est sans carrés, il peut être calculé, à partir de ses $2D$ premières sommes de Newton, en

$$O(M(D) \log(D)).$$

La conclusion reste vraie si toutes les racines de h sont de multiplicité $< p = \text{char}(k)$.

Retrouver un polynôme à partir de ses sommes de Newton en petite caractéristique

Entrée : les premiers termes de $\text{Newton}(h)$.

Sortie : le polynôme h .

$S \leftarrow \text{Newton}(h)$;

$i \leftarrow 1$;

tant que $S \neq 0$ faire

$v_i \leftarrow \text{PolynômeMinimal}(\text{Coeffs}(S))$;

$S \leftarrow S - \text{Newton}(v_i)$;

$i \leftarrow i + 1$;

$t \leftarrow i - 1$;

$v \leftarrow v_1 v_2 \cdots v_t$; // $v_i = \prod_{\substack{h(\gamma)=0 \\ \text{mult}(\gamma) \geq i}} (T - \gamma)$.

retourner v .

Calcul de $f \otimes g$

Lemme

Soit f et g dans $k[T]$. Alors on a la formule :

$$\text{Newton}(f \otimes g) = \text{Newton}(f) \odot \text{Newton}(g),$$

où par \odot on note le produit de Hadamard (terme à terme) de deux séries formelles.

Algorithme

1. calculer les séries $\text{Newton}(f)$ et $\text{Newton}(g)$ à précision $D := \deg(f) \deg(g)$.
2. calculer le produit \odot de ces deux séries.
3. convertir $\text{Newton}(f \otimes g) \longrightarrow f \otimes g$.

Calcul de $f \oplus g$

Lemme

Soit $f, g \in k[T]$ et soit E la série $\exp(T)$.

1. Si $\text{char}(k) = 0$, alors on a la formule :

$$\text{Newton}(f \oplus g) \odot E = (\text{Newton}(f) \odot E) (\text{Newton}(g) \odot E).$$

2. Si $\text{char}(k) = p > 0$, la même formule reste valide modulo T^p .

Étapes de l'algorithme

1. $\text{Newton}(f)$ et $\text{Newton}(g)$ à précision D .
2. produit de Hadamard par E .
3. produit des séries.
4. "division" de Hadamard par E .
5. conversion $\text{Newton}(f \oplus g) \longrightarrow f \oplus g$.

Traces et produit diamant

- Q l'algèbre quotient $k[X, Y]/(f(X), g(Y))$.
- *la trace*, forme linéaire définie sur Q :
trace(A) est la trace de $Q \xrightarrow{\cdot A} Q$.

Lemme [Stickelberger]

La trace de l'endomorphisme de multiplication par A dans Q est égale à

$$\sum_{\substack{f(\alpha)=0 \\ g(\beta)=0}} A(\alpha, \beta).$$

Corollaire

$$\text{Newton}(f \diamond_H g) = \sum_{i \geq 0} \text{trace}(H^i) T^i.$$

Calcul de $f \diamond_H g$

1. on calcule la trace $\in \widehat{Q}$.
2. on calcule les $N = O(D)$ premiers termes de la série Newton($f \diamond_H g$) :

$$\mathcal{L} := [\text{trace}(H), \text{trace}(H^2), \dots, \text{trace}(H^N)].$$

► *projection de puissances.*

3. on récupère $f \diamond_H g$, à partir de \mathcal{L} .

► conversion.

► Goulot d'étranglement : la projection.

Calcul de la trace

- base monomiale de Q :

$$\mathcal{M} = \{x^i y^j \mid 0 \leq i < \deg(f), 0 \leq j < \deg(g)\}.$$

- représentation d'un élément A de Q :
liste de ses coefficients dans la base \mathcal{M} .
- représentation des formes linéaires :

$$\ell \in \widehat{Q} \longleftrightarrow [\ell(\mathbf{m}) : \mathbf{m} \in \mathcal{M}].$$

Lemme

$$\text{trace}(x^i y^j) = \sum_{\substack{f(\alpha)=0 \\ g(\beta)=0}} \alpha^i \beta^j = N_i(f)N_j(g).$$

Complexité *La trace se calcule en*

$$D + O\left(M(\max(\deg f, \deg g))\right).$$

Complexité du produit dans

$$Q = k[X, Y]/(f(X), g(Y))$$

- A, B dans Q .
- Forme particulière de l'idéal définissant Q
 - ▶ AB se calcule en $O(M(D))$.

1. calculer le produit de A et B en $k[X, Y]$.

- ▶ substitution de Kronecker :

$$O(M(D)).$$

2. réduire ce produit modulo $(f(X), g(Y))$.

- ▶ modulo f : $O\left(\deg(g)M(\deg(f))\right)$.
- ▶ modulo g : $O\left(\deg(f)M(\deg(g))\right)$.

Produit transposé et séries génératrices

À $\ell \in \widehat{Q}$, on associe la série génératrice

$$S(\ell) = \sum_{i \geq 0, j \geq 0} \ell(x^i y^j) X^i Y^j.$$

Lemme

La série $S(\ell)$ est rationnelle, de la forme

$$S(\ell) = \frac{N(\ell)}{\text{rec}(f) \text{rec}(g)},$$

*où $N(\ell)$ est un polynôme dans $k[X, Y]$,
de degré $< \deg(f)$ en X et $< \deg(g)$ en Y .*

Complexité du produit transposé

- *Polynôme réciproque de $A \in Q$:*

$$\text{REC}_{m,n} \left(\sum_{\substack{i < m \\ j < n}} A_{i,j} x^i y^j \right) = \sum_{\substack{i < m \\ j < n}} A_{i,j} x^{m-i} y^{n-j}.$$

Lemme *Si ℓ dans \hat{Q} et A dans Q , alors :*

$$\text{REC}_{m,n}(N(A \circ \ell)) = A \cdot \text{REC}_{m,n}(N(\ell)) \quad \text{mod } (f, g).$$

Corollaire

Le produit transposé de $\ell \in \hat{Q}$ et de $A \in Q$ se calcule en $O(M(D))$.

Projection de puissances

[trace(H), ..., trace(H^D)].

- Naïvement : $O(M(D)D)$.
- Pas de bébé / pas de géant (Shoup) :

▲ Structure de Q -module du dual \widehat{Q} :

$$A \circ \ell : Q \rightarrow k$$

$$B \mapsto \ell(AB).$$

le produit transposé de $A \in Q$ et $\ell \in \widehat{Q}$.

- calculer $[1, H, \dots, H^s]$, pour $s = \sqrt{D}$.
- calculer $[H^s \circ \text{trace}, \dots, H^{(s-1)s} \circ \text{trace}]$.
- pour $0 \leq a < s$ et $0 \leq b < s$, calculer

$$\text{trace}(H^{as+b}) = (H^{as} \circ \text{trace})(H^b).$$

► $\underbrace{s \text{ mult. dans } Q}_{O(\sqrt{D}M(D))} + \underbrace{s \text{ prod. tr.}}_{O(\sqrt{D}M(D))} + \underbrace{s^2 \text{ éval } \widehat{Q}}_{D^2}$

► On gagne un facteur \sqrt{D} !

Implantation

- Algorithme implanté en Magma v. 2.9
- Calculs effectués sur **MEDICIS**,
processeur AMD Athlon, 1.5 GB et 1 Ghz.

$$K = GF(10^{30} + 57).$$

degré entrée	150	200	250	300
degré sortie	22500	40000	62500	90000
BoFlSaSc	80	230	244	544
Resultant	3967	12539	30613	63479

Les temps de calcul sont donnés en secondes.

Courbes des complexités

