Détendez vous !

mais...

Ne soyez pas trop paresseux

PAR

Joris van der Hoeven

Sophia Antibes

13 décembre 2002

Plan

Multiplication de séries formelles

Entrée : $\begin{cases} f = f_0 + \cdots + f_{n-1}\, z^{n-1} \\ g = g_0 + \cdots + g_{n-1}\, z^{n-1} \end{cases}$

Sortie : $h = h_0 + \cdots + h_{n-1}\, z^{n-1} = f\, g + O(z^n)$.

Algorithmes classiques de multiplication

- Multiplication naïve en $O(n^2)$.
- Diviser pour régner en $O(n^{\log_2 3})$.
- Multiplication F.F.T. en $O(n \log n \log\log n)$.

Autres opérations sur des séries formelles

Algorithme	Temps	Espace
Multiplication	$n \log n \log\log n$	n
Division	$M(n)$	n
Équations différentielles	$M(n)$	n
Fonctions holonomes	n	n
Composition algébrique	$M(n) \log n$	n
Composition générale	$M(n)\sqrt{n \log n}$	$n \log n$
Composition char. fini	$M(n) \log n$	n
Inversion $\rightarrow$ compostion	$\uparrow$	$\uparrow$

$M(n)$: temps pour la multiplication

Diviser pour régner

Pour n pair, décomposer :

$$\begin{cases} f^{\downarrow} &= f_0 + \cdots + f_{n/2-1}\, z^{n/2-1} \\ f^{\uparrow} &= f_{n/2} + \cdots + f_{n-1}\, z^{n/2-1} \end{cases} \qquad \begin{cases} g^{\downarrow} &= g_0 + \cdots + g_{n/2-1}\, z^{n/2-1} \\ g^{\uparrow} &= g_{n/2} + \cdots + g_{n-1}\, z^{n/2-1} \end{cases}$$

Appliquer récursivement :

$$\begin{aligned} f\,g \;=\; & f^{\downarrow}\, g^{\downarrow} + f^{\uparrow}\, g^{\uparrow}\, z^n + \\ & \left[(f^{\downarrow} + f^{\uparrow})\,(g^{\downarrow} + g^{\uparrow}) - f^{\downarrow}\, g^{\downarrow} - f^{\uparrow}\, g^{\uparrow} \right] z^{n/2} \end{aligned}$$

$g^{\uparrow}$		$f^{\uparrow}\, g^{\uparrow}$
$g^{\downarrow}$	$f^{\downarrow}\, g^{\downarrow}$	
$\times$	$f^{\downarrow}$	$f^{\uparrow}$

$$\square = (f^{\downarrow} + f^{\uparrow})\,(g^{\downarrow} + g^{\uparrow}) - f^{\downarrow}\, g^{\downarrow} - f^{\uparrow}\, g^{\uparrow}$$

Logarithme

$$\log f = \log f_0 + \int \frac{f'}{f}$$

Exponentiation

Pour n pair, supposons

$$\log g - f = O(z^{n/2}),$$

$$\text{avec} \quad \begin{cases} f = f_0 + \cdots + f_{n-1}\, z^{n-1}; \\ g = g_0 + \cdots + g_{n/2-1}\, z^{n/2-1} \end{cases}$$

Alors

$$\tilde{g} = g - (\log g - f)\, g$$

$$\Longrightarrow \log \tilde{g} - f = O(z^n).$$

$\longrightarrow$ algorithme d'exponentiation en $O(M(n))$.

Inversion

L'équation $f \circ g - z = 0$ induit l'itération

$$\tilde{g} = g - \frac{f \circ g - z}{f' \circ g}$$

Position du problème

Entrées :

- $f = f_0 + \cdots + f_{p-1}\, z^{p-1}$ (avec $p \to \infty$) ;
- $g = g_1\, z + \cdots + g_{q-1}\, z^{q-1}$ (avec q fixe) ;
- Un ordre $n \geqslant p$ (avec $n \to \infty$).

Sortie : $h = h_0 + \cdots + h_{n-1}\, z^{n-1}$, telle que

$$h = f \circ g + O(z^n)$$

Algorithme par dichotomie ($p, q, n \in 2^{\mathbb{N}}$)

$$f \circ g = f^{\downarrow} \circ g + (f^{\uparrow} \circ g) \times g^{p/2}$$

Algorithme en temps $O(\frac{p\,q}{n} M(n) \log n)$, car

- $1 + 2 + \cdots + \frac{p\,q}{n}$ multiplications de longueur n.
- $\frac{2\,p\,q}{n}$ multiplications de longueur $n/2$.
- $\frac{4\,p\,q}{n}$ multiplications de longueur $n/4$;
- Etc.
- $p/2$ multiplications de longueur q.

Problème

Entrée : $f_0 + \cdots + f_{n-1}\, z^{n-1}$ et $g_1\, z + \cdots + g_{n-1}\, z^{n-1}$

Sortie : $h_0 + \cdots + h_{n-1}\, z^{n-1}$ avec $h = f \circ g + O(z^n)$

Algorithme de Brent & Kung

Idée : découper $g = g_* + g^*$

$$\begin{aligned} g_* &= g_1\, z + \cdots + g_{q-1}\, z^{q-1}; \\ g^* &= g_q\, z^q + \cdots + g_{n-1}\, z^{n-1}. \end{aligned}$$

Puis écrire :

$$f \circ g = f \circ g_* + (f' \circ g_*)\, g^* + \frac{1}{2}\,(f'' \circ g_*)\,(g^*)^2 + \cdots$$

Calcul des $f^{(n)} \circ g_*$

Par itération direct ou inverse :

$$\begin{aligned} f^{(i)} \circ g_* &= \frac{(f^{(i-1)} \circ g_*)'}{g'_*}; \\ \frac{1}{(i-1)!}\, f^{(i-1)} \circ g_* &= f_{i-1} + i \int \left(\frac{1}{i!}\, f^{(i)} \circ g_* \right) g'_*. \end{aligned}$$

Principe

On considère des séries formelles comme des flots de coefficients. Les coefficients sont calculés un par un et à chaque étape on n'effectue que les calculs strictement nécessaires.

Implantation

Une série formelle f est un algorithme qui ne prend rien en entrée et qui rend son premier coefficient f_0 et la série "reste" $(f - f_0)/z$.

Conséquence importante

On calcule $(f\,g)_n$ dès que $f_0, ..., f_n$ et $g_0, ..., g_n$ sont connus. En particulier, f_{n+1} et g_{n+1} peuvent dépendre de $(f\,g)_0, ..., (f\,g)_n$.

Application

Calcul de l'exponentielle $g = e^f$ d'une série f par

$$g = \int f' \, g$$

Inconvénient

On ne peut utiliser la multiplication F.F.T. ou diviser pour régner.

Idée : anticipation $\longrightarrow$ accélération

Algorithme naïf

g_2	2		
g_1	1	2	
g_0	0	1	2
$\times$	f_0	f_1	f_2

0 $h_0 = f_0\, g_0$.
1 $h_1 = f_0\, g_1 + f_1\, g_0$.
2 $h_2 = f_0\, g_2 + f_1\, g_1 + f_2\, g_0$.

Algorithme détendu

g_2	2		
g_1	1	2	
g_0	0	1	2
$\times$	f_0	f_1	f_2

0 $h_0 = f_0\, g_0$.
1 $h_1 = (f_0 + f_1)\,(g_0 + g_1) - f_0\, g_0 - f_1\, g_1$
2 $h_2 = f_0\, g_2 + f_1\, g_1 + f_2\, g_0$.

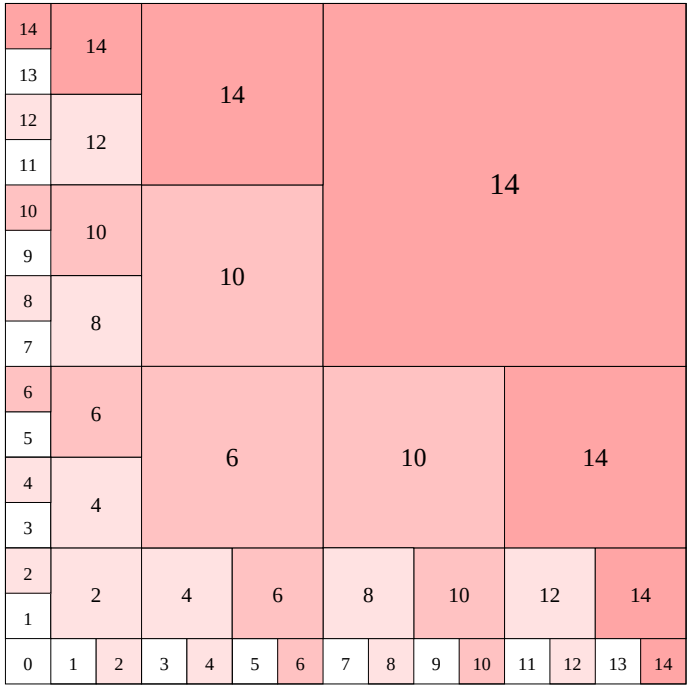
La multiplication diviser-pour-régner est « essentiellement déten-due » : la formule pour h_k ne dépend que de $f_0, \ldots, f_k$ et $g_0, \ldots, g_k$.

Exemple : multiplication à l'ordre 4

- $h_0 = f_0\, g_0$;
- $h_1 = (f_0 + f_1)\,(g_0 + g_1) - f_0\, g_0 - f_1\, g_1$;
- $h_2 = (f_0 + f_2)\,(g_0 + g_2) - f_0\, g_0 - f_2\, g_2$;
- $h_3 = (f_0 + f_1 + f_2 + f_3)\,(g_0 + g_1 + g_2 + g_3) - (f_0 + f_1)\,(g_0 + g_1) - (f_2 + f_3)\,(g_2 + g_3) + f_0\, g_0 + f_1\, g_1 + f_2\, g_2 + f_3\, g_3$;
- $h_4 = (f_1 + f_3)\,(g_1 + g_3) - f_1\, g_1 - f_3\, g_3$;
- $h_5 = (f_2 + f_3)\,(g_2 + g_3) - f_2\, g_3 - f_2\, g_3$;
- $h_6 = f_3\, g_3$.

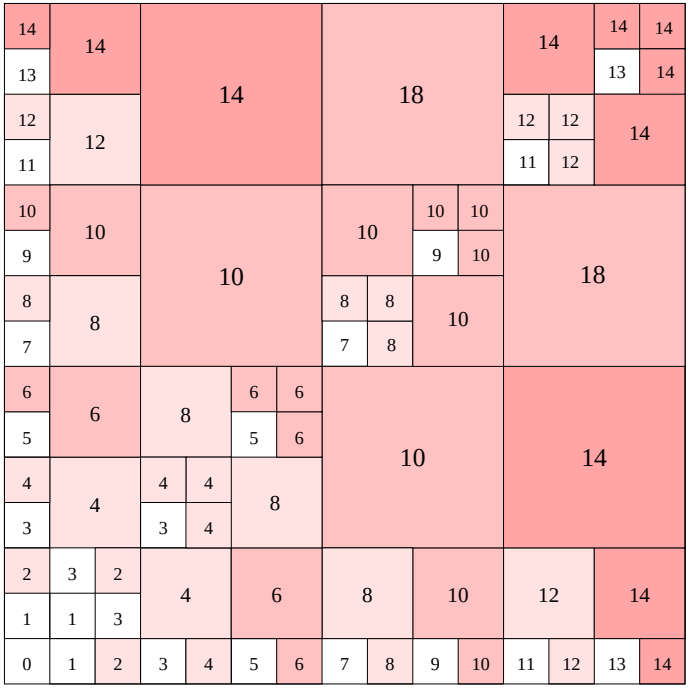
g_3	3	3	3	3
g_2	2	3	2	3
g_1	1	1	3	3
g_0	0	1	2	3
$\times$	f_0	f_1	f_2	f_3

Fast relaxed multiplication

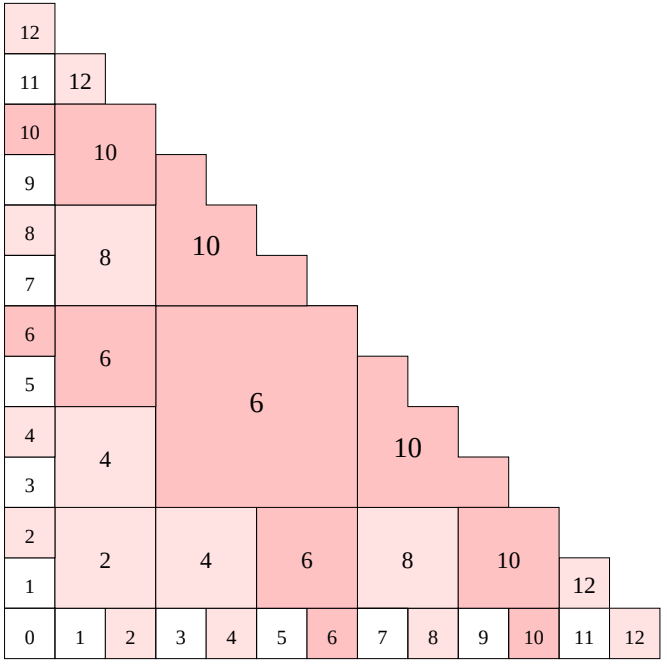


$\longrightarrow$ Algorithme détendu en $O(M(n)\log n)$.

A variant


14 14 14 18 14 14 14
13 13 14
12 12 12 12 12 14
11 11 12
10 10 10 10 10 10 18
9 9 10
8 8 8 8 10
7 7 8
6 6 8 6 6 10 14
5 5 6
4 4 4 4 8
3 3 4
2 3 2 4 6 8 10 12 14
1 1 3
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

Truncated multiplication



Complexité théorique

n	1	2	3	4	5	6	7	8	9	10	100	1000	10000
Naïf	1	3	6	10	15	21	28	36	45	55	5050	500500	50005000
D.P.R.	1	3	5	9	11	15	19	27	29	33	1251	52137	1844937
Rapide 1	1	3	8	10	18	20	37	39	47	49	2938	103693	4458055
Rapide 2	1	3	8	10	18	20	35	37	45	47	1602	27408	411963
Variante 1	1	3	5	8	14	16	22	24	33	35	1904	66515	2535836
Variante 2	1	3	5	8	14	16	22	24	33	35	1176	20311	300794

Composition polynomiale et algébrique

Algorithmes “essentiellement détendues”.

Composition générale

Changer q dans l'algorithme de Brent & Kung pour $n \in 4^{\mathbb{N}}$.

Algorithme	Temps	Espace
Multiplication D.P.R.	$n^{\log_2 3}$	$n \log n$
Multiplication rapide	$M(n) \log n$	n
Division	$D(n)$	n
Équations différentielles	$D(n)$	n
Fonctions holonomes	n	n
Composition algébrique	$D(n) \log n$	n
Composition générale	$D(n)\sqrt{n \log n}$	$n^{3/2} \log n$
Composition char. fini	$D(n) \log n$	$n \log n$
Inversion $\rightarrow$ compostion	$\uparrow$	$\uparrow$

$D(n)$: temps pour la multiplication détendue

Exemple 1 : développement de $\exp(z\,e^z)$

Multiplication	10	20	50	100	200	500	1000	2000	1h
Zealous	0.161	0.985	7.202	27.017	92.36	361.19	1135.4	3403	2135
Naive	0.048	0.282	2.533	11.474	48.86	317.00	1283.8		1670
DAC	0.079	0.309	1.428	4.384	13.19	61.35	1887.4		1025
Fast	0.061	0.331	2.162	7.583	25.10	96.20	307.2	959	4095
Variant	0.077	0.347	1.874	5.938	18.34	67.27	193.8	494	$*$
Truncated	0.047	0.274	1.838	6.782	21.70	98.21	307.5	947	4408

Big float coefficients

Multiplication	10	20	50	100	200	500	1h
Zealous	0.052	0.187	1.294	6.916	50.09	1085.87	686
Naive	0.025	0.072	0.417	2.194	16.62	446.34	845
DAC	0.029	0.101	0.641	3.614	30.45	918.78	758
Fast	0.038	0.125	0.800	4.190	30.96	430.92	767
Variant	0.047	0.155	0.995	5.658	48.78	888.92	703
Truncated	0.026	0.082	0.485	2.308	15.52	342.05	944

Rational coefficients

Exemple 2 : énumération d'alcools

La série génératrice s telle que s_n est le nombre d'alcools de la forme $\mathrm{C}_n\mathrm{H}_{2n+1}\mathrm{OH}$ vérifie

$$s(z) = 1 + z\,\frac{s(z)^3 + 2\,s(z^3)}{3}$$

Multiplication	500	1000	2000	5000	10000	20000	50000	100000	200000	1h
Naive	0.948	2.897	9.541	52.09	198.46	786.5				43312
DAC	0.992	2.603	6.860	24.70	70.24	204.4	873	2624		121561
Fast	0.863	2.101	5.407	20.93	56.25	147.4	547	1355	3370	217087
Variant	0.918	2.055	4.997	16.28	42.10	108.7	411	1014	2480	275967
Truncated	0.766	2.022	5.151	19.03	52.60	145.3	539	1392	3529	203767

Integer coefficients mod 1234577

Multiplication	10	20	50	100	200	500	1000	2000	5000	1h
Naive	0.012	0.026	0.087	0.249	0.850	5.485	32.56	297.57		4018
DAC	0.013	0.032	0.113	0.308	0.922	5.635	30.72	235.50		3583
Fast	0.015	0.037	0.131	0.375	1.185	4.853	21.33	134.54	2611	5759
Variant	0.017	0.043	0.151	0.407	1.221	5.558	29.59	215.59	3519	5119
Truncated	0.012	0.028	0.098	0.276	0.871	4.496	19.95	129.23	2295	5862

Integer coefficients

Exemple 3 : équations fonctionnelles

L'équation différentielle aux différences

$$f(x) = \frac{1}{x}\,(1 + f(x+1) + f'(x)^2) \tag{1}$$

admet une solution formelle unique en x^{-1} :

$$f(x) = \frac{1}{x} + \frac{1}{x^2} - \frac{1}{x^4} - \frac{3}{x^6} + O\left(\frac{1}{x^7}\right)$$

On réécrit (1) pour $f(x) = f(1/z) = g(z)$:

$$g(z) = z\left(1 + g\left(\frac{z}{1+z}\right) - z^4\, g'(z)^2\right) \tag{2}$$

Résultats expérimentaux IV

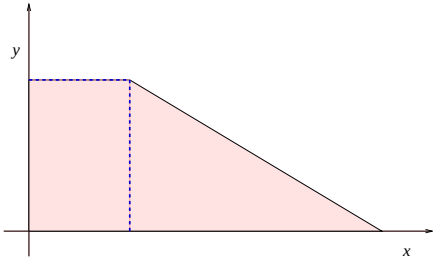
Composition	Multiplication	100	200	500	1000	2000	5000	10000	20000	1h
Naive	Naive	0.537	3.213	43.80	337.1	2647.2				2216
	Fast	1.113	6.187	69.15	459.5	3152.5				2093
	Truncated	0.592	2.857	28.53	169.0	1022.7				3119
Brent&Kung	Naive	0.561	2.065	23.68	96.4	871.1				4148
	Fast	1.067	3.549	34.90	120.0	960.2				4188
	Truncated	0.809	2.650	23.18	75.0	573.8	2905			5628
Fast	Naive	0.406	1.448	8.21	34.2	144.9	1111			8560
	Fast	0.713	2.151	8.13	25.8	83.8	499	1611		16385
	Truncated	0.445	1.366	5.89	19.2	62.9	333	1070	3341	20253

Exemple 4 : é.d.p.

Soit $f \in \mathbb{Q}[[x, y]]$ telle que

$$\begin{aligned} \frac{\partial f}{\partial y} &= \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial^2 f}{\partial x^2}\right)^2; \\ f(x, 0) &= e^x. \end{aligned}$$

Problème : calculer $[x^n\, y^m]\, f$.



Arbres 2-3

Série génératrice f vérifie $f(z) = z + f(z^2 + z^3)$

Arithmétique dense rapide

Benchmarks faussés par mauvaise arithmétique.

F.F.T. généralisée pour “anneaux denses”.

Instabilité numérique

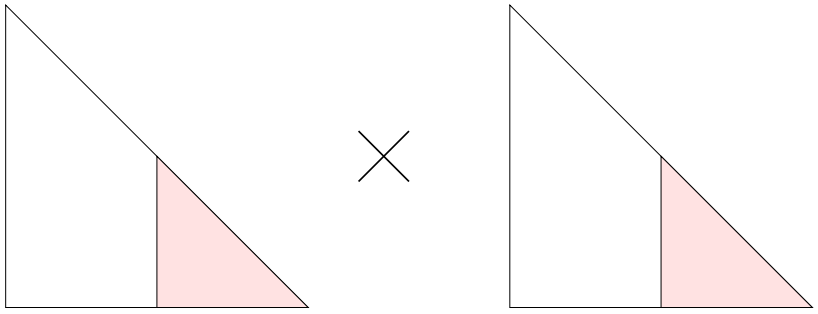
$$\begin{aligned} & (1.000 \cdot 10^0 + 1.000 \cdot 10^{-5}\, z)^2 \\ = \;& 1.000 \cdot 10^0 + 0.000 \cdot 10^{-5}\, z + 1.000 \cdot 10^{-10}\, z^2, \end{aligned}$$

puisque $1.000 \cdot 10^0 + 1.000 \cdot 10^{-5} = 1.000 \cdot 10^0$.

Solution : transformation $z \to \rho\, z$.

Coefficients particuliers II

Problême analogue pour séries bivariées



Solution : multiplication tronquée plus fine ou modulaire.