# Structures de données élémentaires

*Bruno Salvy*

INRIA

# I. Entiers et polynômes, flottants, rationnels

# Calcul sur des entiers

Entiers de taille limitée (à un mot machine)

$\downarrow$

Entiers en précision arbitraire

Opérations souhaitées : $+, -, \times, \div$

Représentation

$$N = \sum_{k=0}^{K} u_k B^k$$

Addition, Soustraction, Multiplication :

comme à la main.

Choix de $B$

Problème de la division

Méthodes modulaires

# Multiplication : l'algorithme de Karatsuba

Deux polynômes $P(x)$ et $Q(x)$

Méthode naïve : $\deg(P) \times \deg(Q)$ multiplications de coefficients.

$$P(x) = P_0(x) + x^k P_1(x), Q(x) = Q_0(x) + x^k Q_1(x)$$

$$P(x)Q(x) = R_0(x) + x^k R_1(x) + x^{2k} R_2(x),$$

avec

$$R_0 = P_0Q_0, \quad R_2 = P_1Q_1, \quad R_1 = P_0Q_1 + P_1Q_0.$$

Idée : 1. $R_1 = (P_0 + Q_0)(P_1 + Q_1) - (R_0 + R_2)$. On a échangé une multiplication contre trois additions. S'applique à n'importe quel anneau de coefficients.

2. On recommence pour les polynômes de degré $n/2$.

# L'algorithme d'Euclide

Dans un anneau euclidien.

$$a_0 = a_1 q_1 + a_2 \Longrightarrow \mathrm{pgcd}(a_0, a_1) = \mathrm{pgcd}(a_1, a_2).$$

On itère, et ça termine.

Note : dans $\mathbb{N}$, pgcd binaire.

Passage aux fractions rationnelles.

# Flottants

Représentation : mantisse, exposant (deux entiers)

$\longrightarrow$ mêmes complexités que les entiers

Conséquence : utilisation de l'algorithme de Newton

# II. Algèbre linéaire

# Matrices

Quelle taille pour quel problème

| Coefficients | Flottants | Entiers | Polynômes |
|---|---|---|---|
| Taille | $180 \times 180$ | $100 \times 100$ | $35 \times 35$ |
| Addition | 40s. | 5s. | 1.s |
| Produit | 1550s. | 80s. | 7s. |
| Déterminant | 840s. | 580s. | 260s. |
| Inverse | 127s. | 1400s. | 1600s. |

# Rôle du calcul formel en algèbre linéaire

- Cas singuliers, instables. Problèmes de décision.
- Calculs de taille modérée à coefficients symboliques.

# Exponentiation binaire

$a^k$ où $k \in \mathbb{N}$ se calcule avec $3/2 \log_2 k$ multiplications en moyenne (dans tout monoïde)

$$a^{11} = ((a^2)^2 a)^2 a$$

Basé sur la décomposition binaire de $k$ ($11 = \overline{1011}_2$).

$$\begin{array}{ccc} 1 & \rightarrow & \text{carré, multiplication} \\ 0 & \rightarrow & \text{carré} \end{array}$$

Remarque : pas encore optimal.

# Exemple

$$u_{n+4} = u_{n+3} + 2u_{n+2} + 5u_{n+1} - 7u_n,$$

$$\begin{pmatrix} u_{n+1} \\ u_{n+2} \\ u_{n+3} \\ u_{n+4} \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -7 & 5 & 2 & 1 \end{pmatrix}}_{A} \begin{pmatrix} u_n \\ u_{n+1} \\ u_{n+2} \\ u_{n+3} \end{pmatrix}$$

$$A^{N-3} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} u_{N-3} \\ u_{N-2} \\ u_{N-1} \\ u_N \end{pmatrix} \qquad (N \geq 3).$$

# Utilisation de l'algorithme de Gauss

Si on peut reconnaître zéro

Déterminant ;
Noyau ;
Inverse ;
...

# III. Séries formelles

# Définition

$$\mathcal{A}[[x]] = \{F(x) = \sum_{k=0}^{\infty} f_k x^k, \quad f_k \in \mathcal{A}\}.$$

$\mathcal{A}$ anneau commutatif $\longrightarrow$ $\mathcal{A}[[x]]$ anneau commutatif

Rq : Pour la suite on prend un sur-corps de $\mathbb{Q}$.

En pratique : séries formelles tronquées

$$F(x) = \sum_{k=0}^{N} f_k x^k + O(x^{N+1}).$$

# Opérations

Complexité en nombre d'opérations dans $\mathcal{A}$

- Somme $F+G$ en $O(N)$ ;
- Produit $F \times G$ en $M(N) = O(N^{\log_2 3})$ (Karatsuba) ;
- Inverse $1/F$, logarithme $\log(1+F)$, exponentielle $\exp(F)$ en $O(M(N))$ ;
- Composition $F(G)$ en $O(NM(N))$.

# La composition

$$F(G(z)), \qquad g_0 = 0.$$

Méthode naïve : $O(N)$ produits de séries.

1. On récrit $F(G)$

   $$F(G(z)) = F_0(G(z)) + G(z)^k [F_1(G(z)) + \cdots]$$

   $$\text{avec } F_i(z) = \sum_{j=0}^{k-1} f_{ki+j} z^j.$$

2. On calcule $G^j(z)$, pour $j = 2, \ldots, k$
   $\rightarrow F_i(G(z))$.

3. On calcule la série par la règle de Hörner.

$N/k$ produits de séries. Optimal pour $k = \sqrt{N}$.

# L'algorithme de Newton

$$\Phi(x, y(x)) = 0$$

$$y_{n+1}(x) = y_n(x) - \frac{\Phi(x, y_n(x))}{\frac{\partial \Phi}{\partial y}(x, y_n(x))}.$$

Le nombre de coefficients corrects double à chaque itération

- Inverse. $\Phi = y^{-1} - F$, $y_{n+1} = 2y_n - Fy_n^2$, $O(M(N))$.
- Logarithme. $y' = F'/F$, $O(M(N))$.
- Exponentielle. $\Phi = \log y - F$, $y_{n+1} = y_n(1 + F - \log y_n)$, $O(M(N))$
- Inversion. $\Phi = F(y) - x$, $y_{n+1} = y_n - [F(y_n) - x]/F'(y_n)$, même coût que la composition.

# Bibliographie

Cohen (Henri). – *A course in computational algebraic number theory.* – Springer-Verlag, 1993.

Davenport (J. H.), Siret (Y.), and Tournier (E.). – *Calcul formel.* – Masson, Paris, 1993, 2ième édition.

Dumas (Philippe) and Gourdon (Xavier). – *Maple — Son bon usage en mathématiques.* – Springer-Verlag, 1997.

Geddes (Keith O.), Czapor (Stephen R.), and Labahn (George). – *Algorithms for Computer Algebra.* – Kluwer Academic Publishers, 1992.

Gomez (Claude), Salvy (Bruno), and Zimmermann (Paul). – *Calcul formel : mode d'emploi. Exemples en Maple.* – Masson, 1995.

Knuth (Donald E.). – *The Art of Computer Programming.* – Addison-Wesley, 1981, 2ième édition, vol. 2: Seminumerical Algorithms.