An output-sensitive variant of the baby steps/ giant steps determinant algorithm

Erich Kaltofen
North Carolina State University
www.kaltofen.net

Matrix determinant definition

$$\det(Y) = \det\left(\begin{bmatrix} y_{1,1} & \cdots & y_{1,n} \\ y_{2,1} & \cdots & y_{2,n} \\ \vdots & & \vdots \\ y_{n,1} & \cdots & y_{n,n} \end{bmatrix}\right) = \sum_{\sigma \in S_n} \left(\operatorname{sign}(\sigma) \prod_{i=1}^{n} y_{i,\sigma(i)} \right),$$

where $y_{i,j}$ are from an *arbitrary commutative ring*, and S_n is the set of all permutations on $\{1, 2, \ldots, n\}$.

Interesting rings: $\mathbb{Z}$, $\mathbb{K}[x_1, \ldots, x_n]$, $\mathbb{K}[x]/(x^n)$

Fast matrix multiplication

Strassen's [1969] $O(n^{2.81})$ matrix multiplication algorithm

$$
\begin{array}{l|l}
m_1 \leftarrow (a_{1,2} - a_{2,2})(b_{2,1} - b_{2,2}) & \\
m_2 \leftarrow (a_{1,1} + a_{2,2})(b_{1,1} + b_{2,2}) & \\
m_3 \leftarrow (a_{1,1} - a_{2,1})(b_{1,1} + b_{1,2}) & \\
m_4 \leftarrow (a_{1,1} + a_{1,2})b_{2,2} & a_{1,1}b_{1,1} + a_{1,2}b_{2,1} = m_1 + m_2 - m_4 + m_6 \\
m_5 \leftarrow a_{1,1}(b_{1,2} - b_{2,2}) & a_{1,1}b_{1,2} + a_{1,2}b_{2,2} = m_4 + m_5 \\
m_6 \leftarrow a_{2,2}(b_{2,1} - b_{1,1}) & a_{2,1}b_{1,1} + a_{2,2}b_{2,1} = m_6 + m_7 \\
m_7 \leftarrow (a_{2,1} + a_{2,2})b_{1,1} & a_{2,1}b_{1,2} + a_{2,2}b_{2,2} = m_2 - m_3 + m_5 - m_7
\end{array}
$$

Problems reducible to matrix multiplication:
linear system solving, determinants [Bunch and Hopcroft 1974],...

Coppersmith and Winograd [1990]: $O(n^{2.38})$

Life after Strassen: bit complexity

Linear system solving $x = A^{-1}b$ where $A \in \mathbb{Z}^{n\times n}$ and $b \in \mathbb{Z}^n$:

With Strassen and Chinese remaindering [McClellan 1973]:

Step 1: For prime numbers $p_1, \ldots, p_k$ Do
Solve $Ax^{[j]} \equiv b \pmod{p_j}$ where $x^{[j]} \in \mathbb{Z}/(p_j)$

Step 2: Chinese remainder $x^{[1]}, \ldots, x^{[k]}$ to $A\bar{x} \equiv b \pmod{p_1 \cdots p_k}$

Step 3: Recover denominators of x_i by continued fractions of $\dfrac{\bar{x}_i}{p_1 \cdots p_k}$.

Length of integers: $k = (n \max\{\log \|A\|, \log \|b\|\})^{1+o(1)}$

Bit complexity: $n^{3.38} \max\{\log \|A\|, \log \|b\|\}^{1+o(1)}$

With Hensel lifting [Moenck and Carter 1979, Dixon 1982]:

Step 1: For $j = 0, 1, \ldots, k$ and a prime p Do

Compute $\bar{x}^{[j]} = x^{[0]} + px^{[1]} + \cdots + p^j x^{[j]} \equiv x \pmod{p^{j+1}}$

1.a. $\widehat{b}^{[j]} = \dfrac{b - A\bar{x}^{[j-1]}}{p^j} = \dfrac{\widehat{b}^{[j-1]} - Ax^{[j-1]}}{p}$

1.b. $x^{[j]} \equiv A^{-1}\widehat{b}^{[j]} \pmod{p}$ reusing $A^{-1} \bmod p$

Step 2: Recover denominators of x_i by continued fractions of $\dfrac{\bar{x}_i^{[k]}}{p^k}$.

With classical matrix arithmetic:

Bit complexity of 1.a: $(n \max\{\log \|A\|, \|b\|\})^{1+o(1)} + n^2(\log \|A\|)^{1+o(1)}$

Total bit complexity: $(n^3 \max\{\log \|A\|, \log \|b\|\})^{1+o(1)}$

Bit complexity of the determinant

With Chinese remaindering: $(n \log \|A\|)^{1+o(1)}$ times matrix multiplication complexity.

Sign of the determinant [Clarkson 92]: $n^{4+o(1)}$ if matrix is ill-conditioned.

Using denominators of linear system solutions [Pan 88, Abbott & Bronstein & Mulders 99]: fast when large first invariant factor.

Using fast Smith form method $n^{3.5+o(1)}(\log \|A\|)^{1.5+o(1)}$ [Eberly & Giesbrecht & Villard 2000]

Baby steps/giant steps algorithm [Kaltofen 1992/2000]

Wiedemann preconditions A and chooses random u and v; then $\det(\lambda I - A) =$ minimal recurrence polynomial of $\{a_i\}_{i=0,1,\ldots 2n-1}$.

Detail of sequence $a_i = u^T A^i v$ computation

Let $r = \lceil \sqrt{2n} \rceil$ and $s = \lceil 2n/r \rceil$.

Step 1. For $j = 1, 2, \ldots, r-1$ Do $v^{[j]} \leftarrow A^j v$;

Step 2. $Z \leftarrow A^r$;

[$O(n^3)$ operations; integer length $(\sqrt{n} \log \|A\|)^{1+o(1)}$]

Step 3. For $k = 1, 2, \ldots, s$ Do ${u^{[k]}}^T \leftarrow u^T Z^k$;

[$O(n^{2.5})$ operations; integer length $(n \log \|A\|)^{1+o(1)}$]

Step 4. For $j = 0, 1, \ldots, r-1$ Do

For $k = 0, 1, \ldots, s$ Do $a_{kr+j} \leftarrow \langle u^{[k]}, v^{[j]} \rangle$.

The state-of-the-art [Kaltofen & Villard ASCM 2001]

Theorem 1
The determinant of an integer matrix can be computed in $O(n^{2.698}(\log \|A\|)^{1+o(1)})$ *bit operations.*
[*Storjohann 2002:* $O(n^{2.38}(\log \|A\|)^{1+o(1)})$.]

Theorem 2
The determinant and adjoint of a matrix over a commutative ring can be computed with $O(n^{2.698})$ *ring additions, subtractions and multiplications.*

Problem 1 (from my 3ECM 2000 talk)
Improve the bit complexity of algorithms for the determinant, resultant, linear system solution, Toeplitz systems, over the integers.

Early termination strategies

Early termination in Newton interpolation [Kaltofen 1986]

For $i \leftarrow 1, 2, \ldots$ *Do*

Pick distinct p_i *and from* $f(p_i)$ *compute*

$$f^{[i]}(x) \leftarrow c_0 + c_1(x - p_1) + \cdots + c_i(x - p_1)\cdots(x - p_i) \equiv f(x) \pmod{(x - p_1)\cdots(x - p_{i+1})}$$

If $f^{[i]}(a) = f(a)$ *for a* ***random*** a *stop.*

End For

Threshold η: In order to obtain a better probability, we require $f^{[i]}(a_j) = f(a_j)$ for **several** random a_j.

Alternative strategy [Emiris 1998, Kaltofen & Lee & Lobo 2000]

For $i \leftarrow 1, 2, \ldots$ *Do*

Pick ***random*** p_i *and from* $f(p_i)$ *compute*

$$\begin{aligned} f^{[i]}(x) \leftarrow c_0 + c_1(x-p_1) + \cdots + c_i(x-p_1)\cdots(x-p_i) \\ \equiv f(x) \pmod{(x-p_1)\cdots(x-p_{i+1})} \end{aligned}$$

If $f^{[i]}(x) = f^{[i-1]}(x)$, *i.e.*, $c_i = 0$ *stop.*

End For

Threshold ζ: In order to obtain a better probability, we require $c_i = c_{i+1} = \cdots = c_{i+\zeta-1} = 0$.

Complications for Chinese Remaindering

Negative values

On-the-fly conversion formula

Prime number distribution [Rosser & Schoenfeld 62]

$c_0 + c_1 p_1 + \cdots + c_{\delta-1} p_1 \cdots p_{\delta-1} \equiv M \pmod{p_1 \cdots p_m}$
where $c_{\delta-1} \neq 0$, $|c_i| < p_{i+1}$, $\operatorname{sign}(c_i) = \operatorname{sign}(M)$.

The probability of false early termination is for **random** $p_i = O(m^\gamma \log m)$ no less than $1 - O(1/m^{\zeta(\gamma-1)-1})$.

FFT-based algorithm [Heindel & Horowitz 71]

Quadruple the number of moduli and perform Lagrangian interpolation. Compare answer with $p_1 \cdots p_{m-\zeta}$.

Adaptive baby steps/giant steps algorithm [Kaltofen 2002]

Detail of sequence $a_i^{[l]} = (u^T A^i v \bmod p_l), 1 \le l \le m$ computation
Let $r = 1$, $Z = A$.

While early termination has not occurred

$r \leftarrow 2r;\ s \leftarrow \lceil 2n/r \rceil;\ m \leftarrow r^2 \log \|A\|;$
Step 1. For $j = 1, 2, \ldots, r-1$ Do $v^{[j,l]} \leftarrow A^j v \bmod p_l;$

Step 2. $Z \leftarrow Z^2$; now $Z = A^r$;
[$O(n^3)$ operations; integer length $(r \cdot \log \|A\|)^{1+o(1)}$]

Step 3. For $k = 1, 2, \ldots, s$ Do ${u^{[k,l]}}^T \leftarrow u^T Z^k \bmod p_l;$
[$O(n^2 \cdot n/r)$ operations; #moduli: $r^2 \cdot \log \|A\|$]

Step 4. For $j = 0, 1, \ldots, r-1$ Do
For $k = 0, 1, \ldots, s$ Do $a_{kr+j}^{[l]} \leftarrow \langle u^{[k,l]}, v^{[j,l]} \rangle \bmod p_l.$

Theorem *[Kaltofen 2002]*
Input: $A \in \mathbb{Z}^{n\times n}$, $b = \log \|A\|$, *threshold* ζ. *Output:* $\det A$
Method: baby steps/giant steps [KV 2001] with early termination (Monte Carlo)
Bit complexity: $(\sqrt{b(b+\zeta+\log|\det A|)}\cdot n^3)^{1+o(1)}$

– Example $\det(A) = O(n^{\boxed{1-\alpha}}b), \zeta = O(1):\ (n^{\boxed{3+1/2-\alpha/2}}b)^{1+o(1)}$

– [Emiris 1998] $(n^{\boxed{4-\alpha}}b)^{1+o(1)}:\ 3+1/2-\alpha/2 < 4-\alpha \Leftrightarrow \alpha < 1$

– [Eberly et al. 2000] $(n^{3+1/2-\alpha/2}b^{\boxed{1+1/2}})^{1+o(1)}$

Theoretical improvements

– by use of Strassen-like fast matrix multiplication (on matrices of dimension $n^{0.45} \times n^{0.45}$

– by blocking à la Kaltofen & Villard 2001 (communicated by V. Pan, Jan 25, 2002)

The curse of soft-O

$\log_2 n < n^{1/3-1/5}$ for $n \geq n_0$: $n_0 \geq 10^{12}$.

$\log_2 n < n^{1/5-1/7}$ for $n \geq n_0$: $n_0 \geq 10^{37}$.

$(\log_2 n)^2 < n^{1/2}$ for $n \geq n_0$: $n_0 \geq 2^{16} = 65536$.

Which algorithm to use when computing an integer determinant?

- Clarkson's when matrix has small orthogonal defect
- Baby steps/giant steps with early termination when determinant is small
- Eberly & Giesbrecht & Villard when invariant factors are wanted
- Pan / Abbott & Bronstein & Mulders when large invariant factor
- Storjohann's high order lifting (???)

Unfortunately, only careful implementation of all of these methods can answer this question.