Polynomial time algorithm

Definition

An algorithm is polynomial time, if there exists a constant k such that the algorithm performs $O(n^k)$ operations on rational numbers whose size is bounded by $O(n^k)$. Here n is the number of bits that encode the input of the algorithm. We say that the algorithm runs in time $O(n^k)$.

Example: Euclidean algorithm

Input: Integers $a \geqslant b \geqslant 0$ not both equal to zero
Output: The greatest common divisor $\gcd(a, b)$

if $(b = 0)$ **return** a
else
    Compute $q, r \in \mathbb{N}$ with $b > r \geqslant 0$ and $a = q \cdot b + r$
    (division with remainder)
    return $\gcd(b, r)$

Analysis

Determinant

Input: $A \in \mathbb{Q}^{n \times n}$
Output: $\det(A)$
if $(n = 1)$
  return a_{11}
else
  $d := 0$
  for $j = 1, \ldots, n$
    $d := (-1)^{1+j} \cdot \det(A_{1j}) + d$
  return d

Analysis

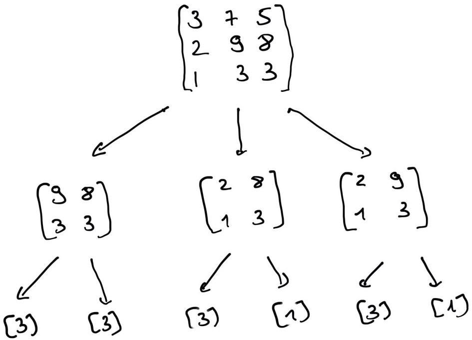


Figure: An example of the recursion tree of the algorithm from Example **??**. The tree corresponds to the run of the algorithm on input $\begin{pmatrix} 3 & 7 & 5 \\ 2 & 9 & 8 \\ 1 & 3 & 3 \end{pmatrix}$.

Gaussian elimination

Input: $A \in \mathbb{Q}^{m \times n}$
Output: A' in row echelon form such that there exists an invertible
$Q \in \mathbb{Q}^{m \times m}$ such that $Q \cdot A = A'$.

$A' := A$
$i := 1$
while $(i \leq m)$
find minimal $1 \leq j \leq n$ such that there exists $k \geqslant i$ such that $a'_{kj} \neq 0$
If no such element exists, then **stop**
swap rows i and k in A'
for $k = i + 1, \ldots, m$
subtract (a'_{kj}/a'_{ij}) times row i from row k in A'
$i := i + 1$

Analysis

Theorem

The Gaussian algorithm runs in polynomial time on input $A \in \mathbb{Z}^{m \times n}$. *More precisely, the rational numbers produced in the algorithm can be maintained to be ratios of sub-determinants of* A' *and are thus of polynomial binary encoding length.*

Matrix multiplication

If we split the matrices A and B into 4 $n/2 \times n/2$ matrices

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \text{ and } B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \tag{6}$$

Then

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} \cdot B_{11} + A_{12} \cdot B_{21} & A_{11} \cdot B_{12} + A_{12} \cdot B_{22} \\ A_{21} \cdot B_{11} + A_{22} \cdot B_{21} & A_{21} \cdot B_{12} + A_{22} \cdot B_{22} \end{pmatrix}.$$

Strassen's algorithm

$$
\begin{aligned}
M_1 &= (A_{11} + A_{22}) \cdot (B_{11} + B_{22}) \\
M_2 &= (A_{21} + A_{22}) \cdot B_{11} \\
M_3 &= A_{11} \cdot (B_{12} - B_{22}) \\
M_4 &= A_{22} \cdot (B_{21} - B_{11}) \\
M_5 &= (A_{11} + A_{12}) \cdot B_{22} \\
M_6 &= (A_{21} - A_{11}) \cdot (B_{11} + B_{12}) \\
M_7 &= (A_{12} - A_{22}) \cdot (B_{21} + B_{22}) \\
&.
\end{aligned}
$$

$$
\begin{aligned}
C_{11} &= M_1 + M_4 - M_5 + M_7 \\
C_{12} &= M_3 + M_5 \\
C_{21} &= M_2 + M_4 \\
C_{22} &= M_1 - M_2 + M_3 + M_6.
\end{aligned}
$$

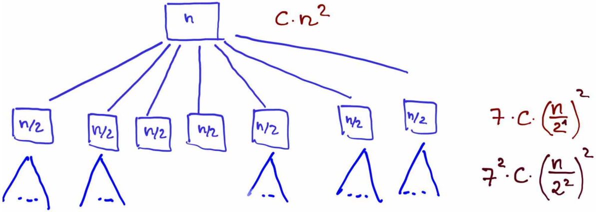
Strassen's algorithm

Input: Two $n \times n$ matrices A and B
Output: $C = \mathrm{FMM}(A, B)$, the product $A \cdot B$

if $n = 1$ return $a_{11} \cdot b_{11}$
else

$M_1 = \mathrm{FMM}(A_{11} + A_{22}, B_{11} + B_{22})$
$M_2 = \mathrm{FMM}(A_{21} + A_{22}, B_{11})$
$M_3 = \mathrm{FMM}(A_{11}, B_{12} - B_{22})$
$M_4 = \mathrm{FMM}(A_{22}, B_{21} - B_{22})$
$M_5 = \mathrm{FMM}(A_{11} + A_{12}, B_{22})$
$M_6 = \mathrm{FMM}(A_{21} - A_{11}, B_{11} + B_{12})$
$M_7 = \mathrm{FMM}(A_{12} - A_{22}, B_{21} + B_{22})$
Compute the matrices $C_{11}, C_{12}, C_{21}, C_{22}$ from $M_1, \ldots, M_7$
return C

Analysis



$$\underline{\text{Total:}} \quad c \sum_{i=0}^{\log_2 n} 7^i \cdot \frac{n^2}{4^i} = c \cdot n^2 \cdot \sum_{i=0}^{\log_2 n} \left(\frac{7}{4}\right)^i$$

$$\le c \cdot n^2 \cdot \left(\frac{7}{4}\right)^{\log_2 n} = c \cdot n^{2+\log_2\left(\frac{7}{4}\right)} = c \cdot n^{2{,}807}$$

Running time

Theorem (Strassen)

Two $n \times n$ matrices can be multiplied in time (number of arithmetic operations) $O(n^{2+\log_2(7/4)})$.

One iteration of the simplex algorithm

Theorem

One iteration of the simplex algorithm requires a total number of $O(m \cdot n)$ operations on rational numbers whose size is polynomial in the input size.