Reminder. Farkas' Lemma.

Given (*) $Ax \le b$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

System (*) has a solution $\iff$

$\forall \lambda \in \mathbb{R}^m_{\geq 0}$ with $\lambda^T \cdot A = 0^T$

one has $\lambda^T \cdot b \geq 0$

Equivalently:

(*) has no solution $\iff$ $\exists \lambda \in \mathbb{R}^m_{\geq 0}$ s.th.

$\lambda^T \cdot A = 0^T$ and $\lambda^T \cdot b < 0$

Thm of DUALITY.

If $\boxed{\begin{array}{l} \max c^T \cdot x \\ Ax \le b \end{array}}$ (P) feasible and bounded, then

$\boxed{\begin{array}{l} \min b^T \cdot y \\ A^T \cdot y = c \\ y \geq 0 \end{array}}$ (D) is feasible & bounded as well and

OPT VALUES COINCIDE.

WEAK DUALITY:

x^*, y^* are (P) & (D) feasible respectively, then $c^T \cdot x^* \le b^T \cdot y^*$.

Strategy: $\forall \varepsilon > 0$ $\exists$ x^*, y^*

(P) & (D) feasible respectively with

$b^T \cdot y^* \le c^T \cdot x^* + \varepsilon$

Proof DUALITY Thm with FARKAS' Lemma.

Let $\delta = \max_{Ax \le b} c^T \cdot x$ $\varepsilon > 0$

$$\begin{bmatrix} c^T x \geq \delta + \varepsilon \\ Ax \le b \end{bmatrix} \approx \begin{bmatrix} -c^T \\ A \end{bmatrix} x \le \begin{bmatrix} -\delta - \varepsilon \\ b \end{bmatrix}$$

is infeasible. FARKAS' LEMMA.

$\exists\ \lambda = \begin{pmatrix} \lambda_0 \\ \lambda_1 \\ \vdots \\ \lambda_m \end{pmatrix} \in \mathbb{R}^{m+1}_{\geq 0}$ s.t.h

$$\lambda^T \cdot \begin{bmatrix} -c^T \\ A \end{bmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \quad (a)$$

$$\lambda^T \cdot \begin{pmatrix} -(\delta+\varepsilon) \\ b \end{pmatrix} < 0 \quad (b)$$

Question.

IS $\lambda_0 = 0$ possible?

ANSWER "NO" otherwise $Ax=b$ INFEASIBLE.

Therefore $\lambda_0 > 0$

OBSERVATION. If λ sat (a) & (b) and $\gamma > 0$ then also $\gamma \cdot \lambda$ satisfies (a) & (b).

W.L.O.G.

$\Rightarrow \quad \lambda_0 = 1.$

Set $\lambda^* = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_m \end{pmatrix}$. (a) & (b)

$\Rightarrow$

a) : $\lambda^{*T} \cdot A = c^T$

b.) : $b^T \cdot \lambda^* < \boxed{\delta} + \varepsilon$

STRATEGY has worked.

λ^* is dual fes. sol

With $b^T \cdot \lambda^* \leq \delta + \varepsilon$.

Let $\varepsilon \to 0$

$\Rightarrow$ Theorem of DUALITY.

$\square$

ALGORITHMS & ANALYSIS

Given $A, B \in \mathbb{Q}^{n \times n}$

Task. Compute $C = A \cdot B$.

FOR $i = 1, \dots, n$

  FOR $j = 1, \dots, n$

    $C_{ij} \boxed{:=} 0$

    FOR $k = 1, \dots, n$

      $C_{ij} \boxed{:=} C_{ij} + a_{ik} \cdot b_{kj}$

Idea: Count number of arithmetic operations $(+, -, *, \%)$

div with remainder

and storage operations.

ARITHMETIC OPERATIONS: $2 \cdot n^3$

STORAGE OPERATIONS: $n^3 + n^2$

Exact counting is sometimes tedious.

$\boxed{RT(A) \leq 4 \cdot RT(B)}$

We analyse with asymptotic growth instead.

Definition. Let $T, f : \mathbb{N} \to \mathbb{R}_{\geq 0}$

i) $T(n) \in O(f(n))$ if $\exists n_0 \in \mathbb{N}_+$ and $d \in \mathbb{R}_+$ s.th.

$$\forall n \geq n_0 : T(n) \leq d \cdot f(n)$$

ii) $T(n) = \Omega(f(n))$ —— " ——

$$\forall n \geq n_0 : T(n) \geq d \cdot f(n)$$

iii) $T(n) = \Theta(f(n))$ if $T(n) = O(f(n))$ AND $T(n) = \Omega(f(n))$

$T(n) = n^3 \cdot \log(n)$

$T(n) = O(n^{3+\varepsilon}) \quad \forall \varepsilon > 0.$

Given $\varepsilon > 0$ find n_0 s.th $\forall n \geq n_0$

$$n^3 \log(n) \leq n^{3+\varepsilon}$$

$$\Leftrightarrow \quad \log(n) \leq n^{\varepsilon}$$

Exercise: FIND n_0

Efficient algorithm, first (problematic) definition.

Algorithm has input. $\sim$ Finite sequence of bits 0/1.

Example. $5 = 101 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$

$\text{Size}(z) = \log(|z|+1)$

ASSYMPTOTICALLY: $\text{Size}(z)$ is number of bits necessary to encode z in binary. $(\Theta(\text{Size}(z)))$

Definition. Let $T, f : \mathbb{N} \to \mathbb{R}_{\geq 0}$

i) $T(n) \in O(f(n))$ if $\exists\, n_0 \in \mathbb{N}_+$ and $\alpha \in \mathbb{R}_+$ s.th.

$$\forall n \geq n_0 : T(n) \leq \alpha f(n)$$

ii) $T(n) = \Omega(f(n))$ —— " ——

$$\forall n \geq n_0 : T(n) \geq \alpha \cdot f(n)$$

iii) $T(n) = \Theta(f(n))$ if $T(n) = O(f(n))$ AND $T(n) = \Omega(f(n))$.

Exemples: $T(n) = n^3 + 10 \cdot n^2 + 1.$

$T(n) = O(n^3)$ Starting from $n_0 = 1$

$\alpha = 12.$

$T(n) \leq 12\, n^3$

$T(n) = \Omega(n^3)$ Starting from $n_0 = 1$

$T(n) \geq \alpha \cdot n^3$ With $\alpha = 1.$

For us: Length of input can be understood of sum of sizes of input numbers.

For a rational number $\alpha = p/q \in \mathbb{Q}$

$\text{size}(\alpha) = \text{size}(p) + \text{size}(q)$, where $\gcd(p,q) = 1$, $p \in \mathbb{Z}, q \in \mathbb{N}_+$

Definition: Algorithm runs in polynomial time, if it executes $O(n^k)$ arithmetic and storage operations, where n encoding length of input.

Example: Matrix Multiplication from before.

$A, B \in \mathbb{Z}^{m \times m}$: $O(n^{3/2})$ where n is total size of the m^2 input numbers.

Why is this problematic?

Algorithm:

INPUT: A sequence of 1's.

$$\underbrace{1, 1, \dots, 1}_{n}$$

$S := 2$

For $i = 1, \dots, n$

$S := S^2$

Number of arithmetic operations is $O(n)$

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

$$(((2)^2)^2)^2 = 2^{2^3}$$

$$\vdots$$

END: $S = 2^{2^n}$ IN BINARY $1\underbrace{0 \cdots\cdots 0}_{2^n - 1}$

Exponential Encoding length

Definition: (Polynomial-time alg)

Algorithm runs in polynomial time, if $\exists\, k \in \mathbb{N}_+$ (constant) s.t.

Alg. performs $O(n^k)$ arithmetic operations on rational numbers of size bounded by $O(n^k)$

Where n is total binary encoding length of input.

Gaussian Elimination:

INPUT: $A \in \mathbb{Z}^{m \times n}$

OUTPUT: $A' = \left(\begin{array}{c|c} \nabla & \square \\ \hline 0 & 0 \end{array}\right)$ FULL RANK

S.th. $\exists$ Permutation matrices $P_1 \in \{0,1\}^{m \times m}$

$P_2 \in \{0,1\}^{n \times n}$

And $L \in \mathbb{Q}^{m \times m}$ lower triangular $L = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ * & & 1 \end{bmatrix}$ with 1's on diagonal and

$$L \cdot P_1 \cdot A \cdot P_2 = A'$$

FOR NOW: ASSUME P_1 & P_2 have been pre-computed. (INVARIANT $a'_{ii} \neq 0$ is pivot element)

$A' := A$

FOR $i = 1, \dots, m$

  For $k = i+1, \dots, m$

    FOR $j = i, \dots, n$

$$a'_{kj} = a'_{kj} - \frac{a'_{ki}}{a'_{ii}} \cdot a'_{ij} \in \mathbb{Q}$$

$$A' = \begin{bmatrix} \nabla & \square & \\ 0 & \begin{matrix} a'_{ii} & \cdots & a'_{in} \\ \vdots & & \\ a'_{mi} & \cdots & a'_{mn} \end{matrix} \end{bmatrix}$$

$$\begin{array}{ccc} \neq 0 & & \\ \boxed{a'_{ii}} \cdots a'_{in} & \Big| & * \dfrac{a'_{ki}}{a'_{ii}} \\ a'_{ki} \cdots a'_{kn} & & - \end{array}$$

Gaussian elimination performs $O(m^2 n)$ arithmetic operations.

How large can size of rational numbers in course of ALGORITM BECOME?