April 16, 2025 12:23 PM

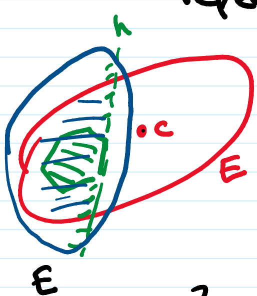
Linear Programming

Simplex method

Ellipsoid Method (Khachiyan '79)

first polytime alg'm
but slower than simplex in practice

iterative method to find a feasible sol'n
maintain ellipsoid E enclosing feasible region



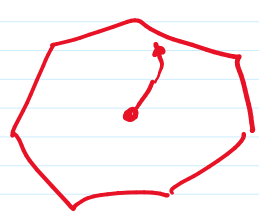
repeat {
  if center c of E is feasible
    return c
  else find constraint h (halfspace) violated by c
    $E \leftarrow$ a smaller ellipsoid that encloses $E \cap h$
}

Claim volume of E decreases exponentially

$\Rightarrow$ # iterations $\leq$ polynomial in n & m & $\log U$

Interior-Point Methods (Karmarkar '84, ..., Lee-Sidford '14, ...)

polynomial & more practical



NP-Hardness

how to prove a problem is ~~hard~~ can't be solved in polytime?

idea: by reduction

To show: if L could be solved in polytime, then some well-known hard problem L_0 could be solved in polytime — L_0 polytime-reduces to L

contradicting hypothesis that L_0 can't be solved in polytime

What choice for L_0?

Problem (SAT) Given Boolean formula F (in CNF form) in vars $x_1, \ldots, x_n$,

decide $\exists$ assignment of $x_1, \ldots, x_n$ s.t. $F(x_1, \ldots, x_n)$ is true

e.g. $F = (x_1 \vee x_2 \vee x_3) \wedge \overbrace{(\bar{x}_1 \vee x_2 \vee x_4)}^{\text{clause}} \wedge (\bar{x}_2 \vee x_3 \vee x_4) \wedge (\bar{x}_1 \vee \bar{x}_3 \vee \bar{x}_4)$ "3CNF"

yes e.g. $x_1 = 1$, $x_2 = 1$, $x_3 = 1$, $x_4 = 0$

brute force: $\sim O(2^n)$ time

k-SAT $\sim O\left(\left(2 - \frac{2}{k}\right)^n\right)$ time

Hypothesis SAT can't be solved in polytime.
(equiv. to Conjecture that "$P \neq NP$")

Rmk: Why SAT?

most problems of the form $\exists \ldots$ s.t. $\ldots$ ("NP problems") reduces to SAT ("Cook-Levin Thm")

e.g. $\exists$ 3-coloring

$\exists$ vertex cover of size $\leq K$

"Definition" L is NP-hard iff SAT polytime-reduces to L.

(L is NP-complete iff SAT polytime-equiv to L
i.e. reduce in both dir.)

Ex 1 SAT polytime-reduces to 3SAT.

Pf: rewrite $\alpha_1 \vee \dots \vee \alpha_k$ $(k > 3)$

$a \to b$

$\bar{a} \vee b$

as
$$
\begin{aligned}
&(y_1 \to \alpha_1 \vee \alpha_2) && (\bar{y}_1 \vee \alpha_1 \vee \alpha_2) \\
&\wedge (y_2 \to y_1 \vee \alpha_3) && \equiv \wedge (\bar{y}_2 \vee y_1 \vee \alpha_3) \\
&\quad \vdots && \quad \vdots \\
&\wedge (y_{k-3} \vee \alpha_{k-1} \vee \alpha_k) && \wedge (y_{k-3} \vee \alpha_{k-1} \vee \alpha_k).
\end{aligned}
$$

Ex 2 0-1 Integer Linear Prog. (ILP)

3SAT polytime-reduces to 0-1 ILP.

Pf: rewrite $x_1 \vee \bar{x}_2 \vee x_3$

as $x_1 + 1 - x_2 + x_3 \geq 1.$

$\square$

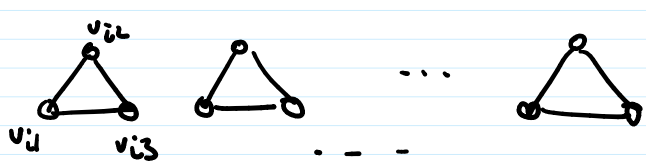
Ex 3 Indep Set: given graph $G = (V, E)$, integer K,
decide $\exists I \subseteq V$, $|I| \geq K$,
$\forall u, v \in I \Rightarrow uv \notin E$.

(Cook '71)

3SAT polytime-reduces to Indep-Set.

Pf: Given 3CNF formula F with n vars,
m clauses $C_1, \dots, C_m$
$C_i = \alpha_{i1} \vee \alpha_{i2} \vee \alpha_{i3}$

Construct graph $G = (V, E)$, K,



create m triangles $v_{i1} v_{i2} v_{i3}$, $i = 1..m$

when $\alpha_{ij} = \overline{\alpha_{i'j'}}$, add edge $v_{ij} v_{i'j'}$.

$K = m$

$\vdots$