Anna Lubiw
Univ. Waterloo

Optimization problem with linear inequalities.

variables $x_1 \cdots x_d$ in d-dimensions.

$$\max \quad c_1 x_1 + c_2 x_2 + \cdots + c_d x_d$$

$$\begin{aligned} \text{s.t.} \quad & a_{11} x_1 + a_{12} x_2 + \cdots + a_{1d} x_d \le b_1 \\ & \vdots \qquad\qquad\qquad\qquad\qquad\quad \vdots \\ & a_{n1} x_1 + \cdots + a_{nd} x_d \le b_n \end{aligned}$$

i.e. $\max \ cx$ c $1 \times d$ vector

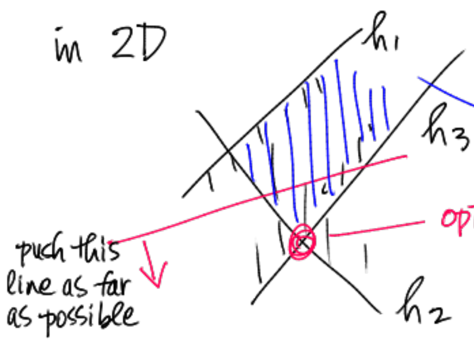
$Ax \le b$ x $d \times 1$ vector

A $n \times d$ matrix

b $n \times 1$

---

in 2D



each constraint $a_1 x_1 + a_2 x_2 \le b$ is a half-space

feasible region

opt. soln may not be unique.

So long as the feasible region is non-empty and bounded ~~the~~ there is an opt. solution ~~is~~ at a vertex. = pick d of inequalities and set to equality

This gives a stupid algorithm - try all $\binom{n}{d}$ vertices. — $O(n^d)$ alg.

- which are feasible?
- which gives max. obj. value.

Applications.

planning menus. Have n nutrients. Need amount b_i of nutrient i.

Have d foods, food j has cost c_j and amount a_{ij} of nutrient i.

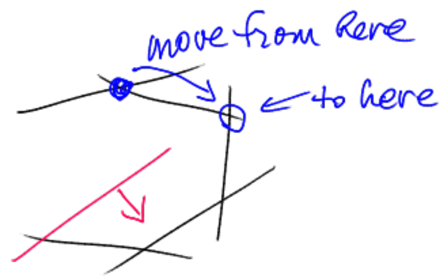
$\min \ cx$ x_j — cost food j

$Ax \ge b$

History 40's, 50's

Dantzig – simplex method '40's.

- spurred development of computers
- geometrically, it walks from one vertex of feasible region to adjacent one



simplex "pivot" rule – which ineq. to remove & which one to add.

- for almost all simplex pivot rules we know examples where it takes exponenial time.

OPEN – is there some pivot rule that gives poly. time?

related to Hirsch conjecture: diameter of convex polyhedron is $\leq n-d$ (n = # inequalities, d = dimension) Disproved in 2012.

but there could still be polynomial (even linear) bound.

Simplex method very good in practice.

poly. time algs. for linear programming:

'80 – Katchian – ellipsoid method.

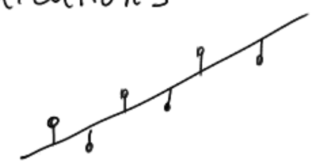
'84 – Karmarkar – interior point method.

Operate on bit representations of numbers.

OPEN: alg. that uses # arithmetic operations poly. in n and d.

- 70's & 80's — linear prog. in small dim.
  $d=2$, $d=3$

Applications.

- 

  find best line fitting pts

- de Berg et al. — whether a cast can be removed from a mold — 3D linear programming.

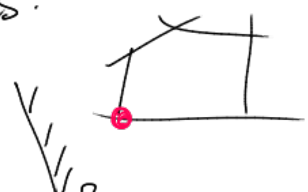
Megiddo '83 $O(n)$ for fixed d
actually $O(2^{2^d} \cdot n)$

---

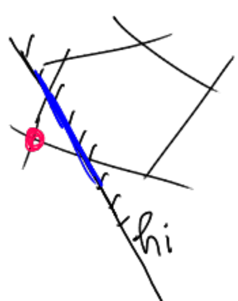
today's topic: Seidel's Randomized Incremental LP Alg.
idea: add half-planes one by one, updating opt. soln v (vertex)

Add h_i

2 cases:

① 

$v \in h_i$ — no update.

② 

$v \notin h_i$ new opt. will lie on l_i — line of h_i

So solve 1-dimensional LP problem along line l_i.

1D LP

$$\max \ x$$
$$x \le 2 \quad \text{soln}$$
$$x \le 5$$
$$-1 \le x$$

find lowest upper bound on x.

$O(i)$

Algorithm $LP_2(H)$ $H = \{h_1 \cdots h_n\}$

1. take random ordering $h_1 h_2 \cdots h_n$
2. $v \leftarrow$ point at infinity.
3. for $i = 1 \ldots n$ /* add h_i
4. if $v \notin h_i$ then
5. $v \leftarrow \underbrace{LP_1(\{h_1, h_2 \cdots h_{i-1}\} \cap \ell_i)}_{\text{1D LP — solve in } O(i)}$ line of h_i

---

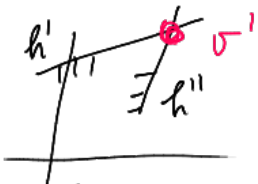
Worst case: $O(n^2)$

Can this worst case happen? Yes. exercise.

<u>Expected run-time.</u>

use Backwards Analysis.

After adding h_i suppose opt. is vertex v' at intersection of h', h''



— we have i lines

halfplane h_i is equally likely to be any one of them.

We did work for h_i (in line 5 of alg) only if $h_i = h'$ or h''.

Prob $\{h_i = h' \text{ or } h_i = h''\} = \frac{2}{i}$ because

Expected total work in line 5 $\sum_{i=1}^{n} \frac{2}{i} O(i) = O(n)$

In higher dimensions

$\frac{2}{i}$ becomes $\frac{d}{i}$ because it takes d hyperplanes to specify a vertex.

$T_d(n) = T_d(n-1) + \frac{d}{n} O(T_{d-1}(n))$ Sol<u>n</u> is $T_d(n) = O(d!\,n)$