Linear Programming

"program" as in "exercise program" or "spending program", not "C program"

optimization problem with linear inequalities

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

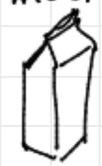
$$\begin{aligned}
\max \quad & c_1 x_1 + c_2 x_2 + \cdots + c_d x_d \\
\text{s.t.} \quad & a_{11} x_1 + a_{12} x_2 + \cdots + a_{1d} x_d \le b_1 \\
& \vdots \\
& a_{n1} x_1 + \cdots + a_{nd} x_d \le b_n
\end{aligned}$$

i.e. $\max\ cx$

$Ax \le b$

c $1 \times d$ vector
x $d \times 1$ vector
A $n \times d$ matrix
b $n \times 1$

An application: planning

	apple	brocolli		milk
d foods			$\dots$	
	1	2		d
each with cost	c_1	c_2	$\dots$	c_d

n nutrients	protein	vitamin D		
	1	2	$\dots$	n
each with daily requirement	b_1	b_2	$\dots$	b_n

a_{ij} – amount of nutrient i in food j

Buy food to meet daily requirements, min cost

$$\min cx$$
$$Ax \geq b$$

variables $x_1 \dots x_d$

x_j = amount of food j to buy.

picture in 2D

Straightforward algorithm:

try all vertices, see which gives max

From last day: this is the dual problem to Convex Hull and can be solved by same algorithms

$O(n \log n)$ in 2D, 3D
$O(n^{\lfloor d-1/2 \rfloor})$ for $d \geq 4$

History

early history 40's, 50's

Dantzig – simplex method in '40's

– spurred development of computers

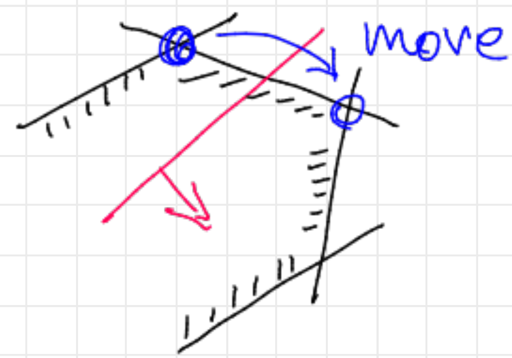
Simplex Method

– geometrically – walk from one vertex of the feasible region to an adjacent one

– Simplex Pivot Rule:

which inequality to remove,

which one to add



History

But the simplex rule is very good in practice.

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

Related question:

Given initial vertex s and final vertex t, how many edges on best path $s \to t$?

diameter of polyhedron = worst case over s, t.

Hirsch conjecture: diameter of convex polyhedron is $\leq n - d$

$n = \#$ inequalities

$d =$ dimension

Disproved in 2012, $d=43$

But there could still by polynomial (even linear) bound.

History

Polynomial time algorithms for Linear Programming:

'80 – Katchian – ellipsoid method
'84 – Karmarkar – interior point method

These operate on bit representations of numbers

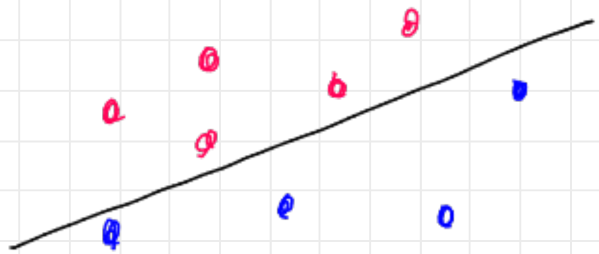
OPEN: an algorithm that uses
arithmetic operations polynomial in n, d

"strongly polynomial time"

Linear Programming in Small Dimensions

Applications

1. Separating red and blue points by a line



$y = ax + b$ variables a, b

$p_y \le a p_x + b \quad \forall (p_x, p_y)$ blue

$p_y \ge a p_x + b \quad \forall$ " red

Solve for a, b (just feasibility, no max.)

... and can check red below, blue above ... vertical line ...

Can also ask for strict separation

$p_y < a p_x + b \implies p_y \le a p_x + b + \delta$

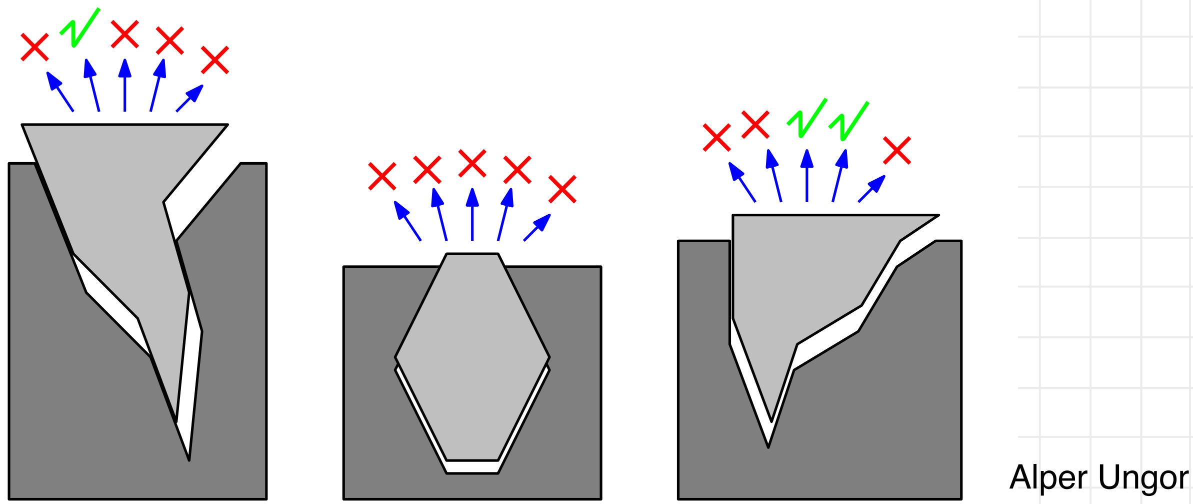
δ a variable, $\delta \ge 0$

maximize δ.

2. Casting (from de Berg et al.)

Make a 3D object in a mold

picture in 2D



Pour liquid into a mold, harden,
then remove by straight line motion in some direction.

Find a direction that works

For given top face – can express as LP.

Try all top faces.

Linear programming in fixed dimension
can be done in time $O(n)$

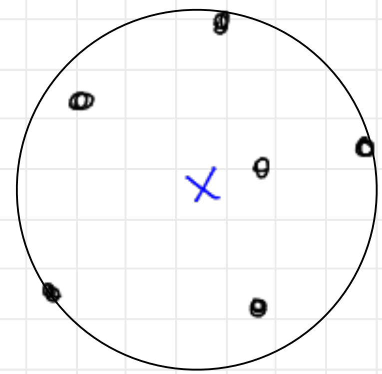
Megiddo 1984
but dependence on d is bad $O(2^{2^d} n)$

Randomized Incremental Algorithm in 2D, Seidel

Smallest enclosing disc

Not a linear programming problem, but amenable to the same approach

Given points $p_1 \cdots p_n$ in $\mathbb{R}^d$
find smallest enclosing sphere.



smallest disc in 2D

This is a facility location problem –
the center of the disc
minimizes the max distance to all points.

Quadratic programming.

Megiddo's approach gives $O(n)$ but bad constant

Randomized Incremental Approach, Welzl