Boyer-Moore — like naive alg. but you match right to left & not left to right, & use info you gain in later steps
- see text.

Rabin-Karp — compute a "hash" function on each window (length m) of text & check only those "windows" which hash to same value as pattern.

4/25/88

Linear Programming

Standard form of LP problem

n variables $x_1, \ldots, x_n$
m inequalities

Maximize $Z = c_1 x_1 + \cdots c_n x_n$ $\qquad Z = \bar{c} \cdot \bar{x}$

Subject to
$$\left.\begin{array}{l} a_{11}x_1 + \cdots a_{1n}x_n \le b_1 \\ \vdots \\ a_{m1}x_1 + \cdots a_{mn}x_n \le b_m \end{array}\right\} \quad A\bar{x} \le \bar{b}$$

and $x_i \ge 0 \quad \forall i$

e.g.

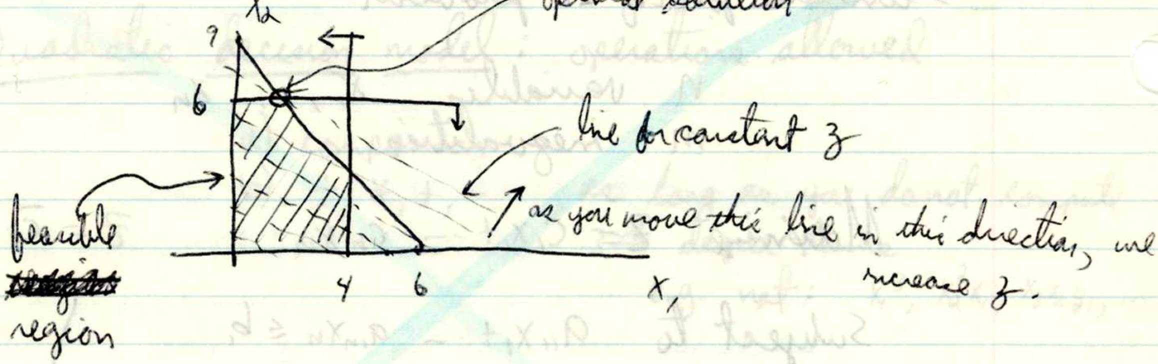
X_1 = amount of product 1 (belts)
X_2 = amount of product 2 (shoes)

maximize $z = 3x_1 + 5x_2$ (profit)

$x_1 \le 4$ (he has 4 belt buckles)
$2x_2 \le 12$ (he has 12 heels)
$3x_1 + 2x_2 \le 18$ (belts require 3 pieces of leather, shoes 2)
only 18 pieces, total units

$x_1 \ge 0,\ x_2 \ge 0$

graphical method:



Now, draw line for constant $z = 3x_1 + 5x_2$

So, optimal solution occurs at (2,6) for $\underline{\underline{z = 36}}$

Note: in general the feasible region is an n-dim. region. Also, for any two points in the region, every point in a straight line from one point to the next will be in the feasible region

e.g. for $(x_1 \_ x_n), (x_1' \_ x_n')$

$\lambda(x_1 \_ x_n) + (1-\lambda)(x_1' \_ x_n')$ for $0 \le \lambda \le 1$

is in feasible region (this is convexity)

Note: if we restrict the problem to integer solutions, it's NP-complete. (I think it's NP-hard)

Variations:

① minimize z (= maximize $-z$)
② $\geq b_i$ (mult by -1 to get $\leq$)
③ $= b_i$ (convert to $\leq b_i$ & $\geq b_i$ & apply ②)
④ x_i unrestricted — (let $x_i = x_i' - x_i''$ where $x_i' \geq 0, x_i'' \geq 0$)
⑤ Decision version:
is feasible region non-empty?

*

Simplex algorithm

— Average time $\approx O(m^3)$ (empirical)

Worst case $= O(m^n)$

— 1979 - "Ellipsoid algorithm" Khacian, Shor

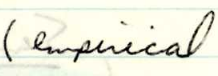
worst case $= O((m+n)n^4 \cdot L)$

L = length of input in bits
(matrix A, vector b & c)
— i.e. dependent on precision.

Karmarkar — efficient & competitive interior point method

$\therefore LP \in P$

Reduce Net. flow to LP.



$c(u,v)$ $f(u,v)$

$f(u,v) \leq c(u,v)$

$f(u,v) = -f(v,u)$

flow into u = flow out of u $u \neq s,t$

maximize flow out of $s = \sum_u f(s,u)$

Simplex Method

def: feasible region = set of all pts. satisfying all constraints

def: corner pt. = intersection of n hyperplanes, where hyperplane results from replacing inequality by an equality.

$$\# \text{ of corner points} \le \binom{m+n}{n} \le 2^{m+n}$$

we have $m+n$ hyperplanes & in n-space we need the intersection of n hyperplanes to define a pt. Thus we have $(m+n)$ choose n pts. $= \binom{m+n}{n}$

At a corner pt at least n of the inequalities are satisfied with equality.

Simplex method

1. start at a feasible corner pt., e.g. $\bar{x} = 0$
2. while $\exists$ an adjacent feasible corner pt. with better z, move to it.
3. Stop, current point is optimal.

Look at previous problem:

$$\begin{array}{rcl}
z - 3x_1 - 5x_2 & & = 0 \\
x_1 & + x_3 & = 4 \\
2x_2 & + x_4 & = 12 \\
3x_1 + 2x_2 & + x_5 & = 18
\end{array}$$

* See paper

~~basic~~ zero ⟷ basic ~~zero~~

note: x_3, x_4, x_5 are slack variables so we change inequalities to equalities.

leaving zero variable = variable whose increase will cause greatest increase in z
e.g. here it's x_2

leaving basic variable = examine equations. How much can you increase leaving zero var. before the corresp. basic variable becomes negative? which equation which ... ...

see notes for matrix simplex method.

running time per step $= O(m(m+n))$
row calc.

\# steps = $O(m)$ (empirically)

time $\approx O(m^2(m+n))$

4/27/88

Convex Hulls in 2 dimension

Convex Hull of a set of pts. S is the smallest ~~(by area)~~ convex figure that contains all the points of S.

If the set of points is finite, then it can be shown that (the convex hull) is a convex polygon (in 2-d) or a convex polyhedron (in 3-d or higher)

note: convex – if two points are in convex hull, then every point in between are in convex hull.

– convex hull vertices are also called extreme points

Computation of a convex hull of S ⇒ output the vertices in some order (e.g. clockwise)

convex hull solution ⇒ some ordering of the answer.

Ideas

a) take a point (e.g. one with lowest x-coordinate) and "wrap" a line around the set of points

b) keep a "working convex hull" of i points & update it by adding the $(i+1)^{th}$ points

c) divide & conquer.

observation: A point $\in CH(S)$ (i.e. in the interior of C.H.) if there are 3 pts. in S whose triangle contains this point

$\Rightarrow$ straightforward algorithm

$\binom{n}{3}$ tuplets of pts. For each tuplet find the points that are inside it and throw them away.

Theorem: $\Omega(n \lg n)$ time is required to compute the convex hull.

Quadratic decision model: operations allowed

a) comparison
b) $\div, \times, +, -$ as long as you do not compute 3^{rd} degree terms
e.g. not: $x_1^3,\ x_2^2 x_1,\ x_3 x_2 y_1, \dots$

reduce sorting to convex hull

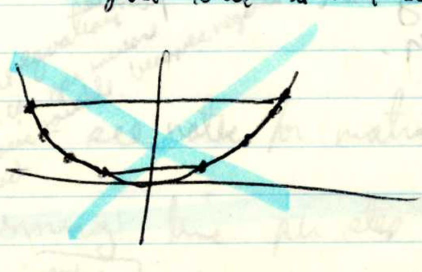
obs 2: Suppose you are given n #'s

$\{x_1 \dots x_n\}$

produce n points (x_i, x_i^2)

$\{(x_1, x_1^2), (x_2, x_2^2) \dots (x_n, x_n^2)\}$

give this to the alg. that finds the convex hull



the (clockwise) output of the convex hull will be a permutation of sorted order — must take $\Omega(n \lg n)$ time