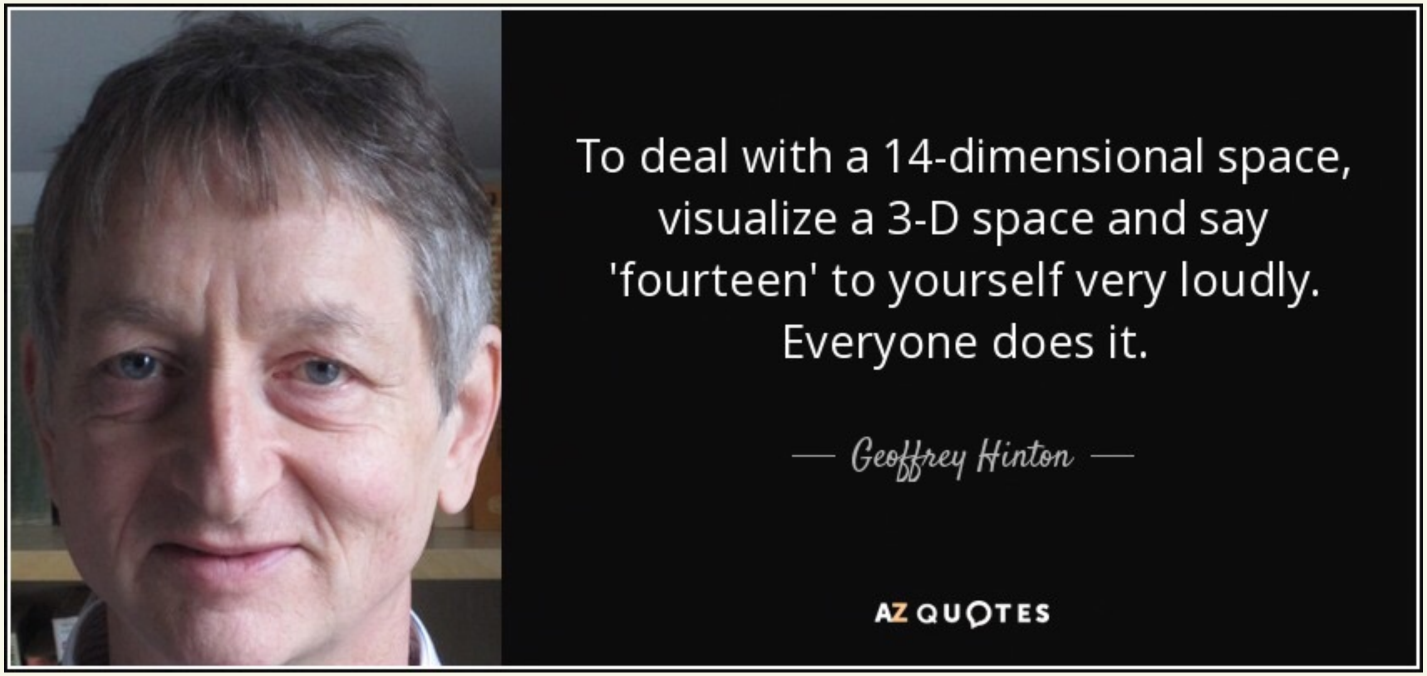
Algorithms - Spring '25

LPs - Duality

Recap:

- Make sure to sign up for HW8 grading slot!
- Practice final is posted
- CIFs are live, & feedback is appreciated

A moment of honesty to start lecture:



Linear optimization: Example

n foods, m nutrients

Let a_{ij} = amount of nutrient i in food j

r_i = requirement of nutrient i

x_j = amount of food j purchased

c_j = cost of food j

Goal: Buy food so you satisfy nutrients while minimizing cost

Can view as matrix $\longrightarrow$

$$A = \begin{array}{cc} & j \\ i & \begin{bmatrix} & & \\ & a_{ij} & \\ & & \end{bmatrix} \end{array}$$

$\vec{r} = (r_1, r_2, \ldots r_m)$

$\vec{x} = (x_1, \ldots, x_n)$

$\vec{c} = (c_1, \ldots, c_n)$

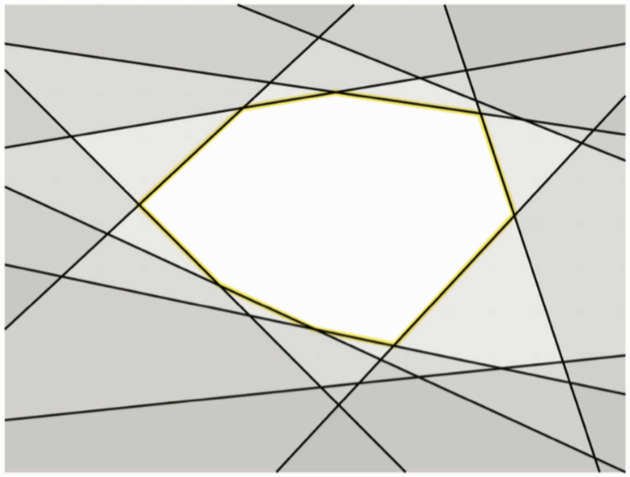
So: minimize

s.t.

In general, get systems like this:

$$
\begin{aligned}
\text{maximize } & \sum_{j=1}^{d} c_j x_j \\
\text{subject to } & \sum_{j=1}^{d} a_{ij} x_j \leq b_i \quad \text{for each } i = 1 \,..\, p \\
& \sum_{j=1}^{d} a_{ij} x_j = b_i \quad \text{for each } i = p+1 \,..\, p+q \\
& \sum_{j=1}^{d} a_{ij} x_j \geq b_i \quad \text{for each } i = p+q+1 \,..\, n
\end{aligned}
$$

Geometric picture:



A two-dimensional polyhedron (white) defined by 10 linear inequalities.

History

Dates back to 1800's where studied by Fourier.

By 1940's: serious study, due to business/optimization demand

- Not known to be NP-Hard
  (Karp actually listed it as key open question in original paper on NP-Hardness)

Canonical form:

Avoid having both $\leq$ and $\geq$.

Why?

So get something more like our first example:

$$\text{maximize} \sum_{j=1}^{d} c_j x_j$$

$$\text{subject to} \sum_{j=1}^{d} a_{ij} x_j \leq b_i \quad \text{for each } i = 1..n$$

$$x_j \geq 0 \quad \text{for each } j = 1..d$$

Or, given a vector $\vec{c}$, matrix A + vector $\vec{b}$:

Anything can be put into canonical form:

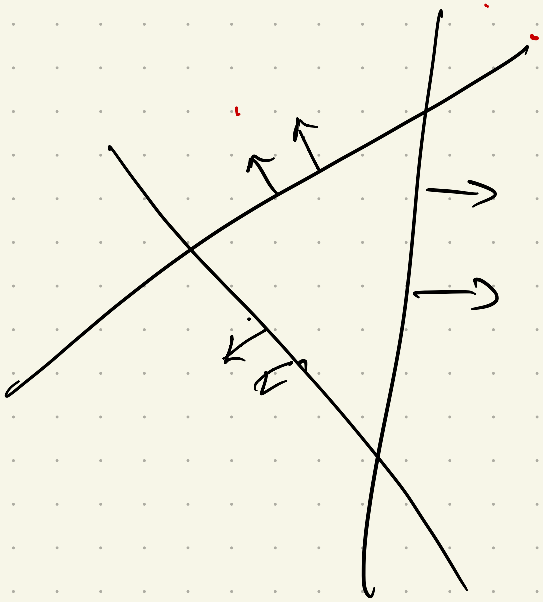
The reduction:

1. Avoid $=$:

2. Avoid $\geq$:

How could these not have a solution?

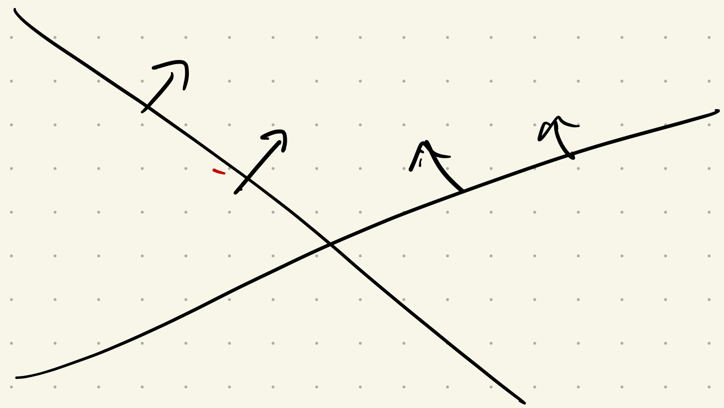
2 ways:



or

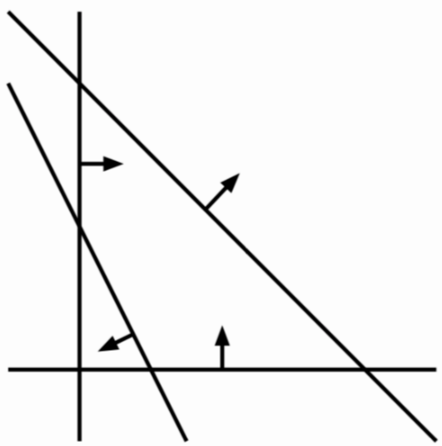
maximize $x_1 + x_2$

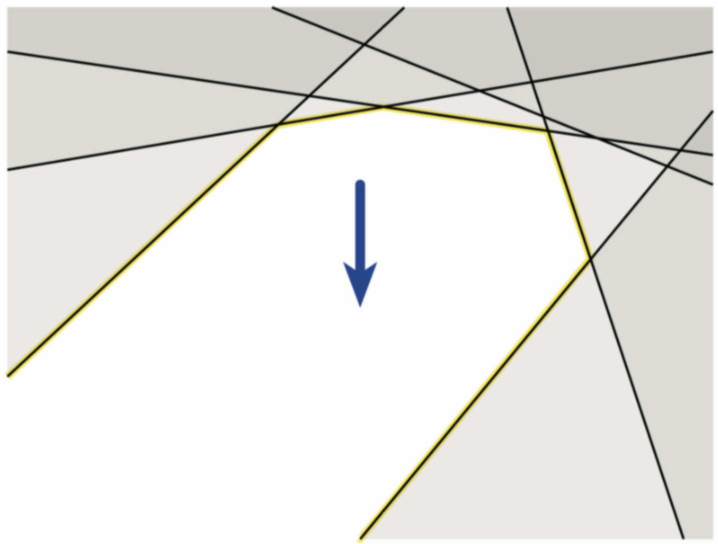
s.t.



Better pictures (still 2-d):

$$\begin{aligned} \text{maximize } & x - y \\ \text{subject to } & 2x + y \leq 1 \\ & x + y \geq 2 \\ & x, y \geq 0 \end{aligned}$$





Note:

1. Multiplying by -1 turns any maximization problem into a minimization one:

why?

2. Can turn inequalities into equalities via slack variables:

$$\sum_{i=0}^{n} a_i x_i \le b \implies$$

(3) Can change equalities into inequalities, also!

$$\sum_{i=1}^{n} a_i x_i = b$$

$$\Downarrow$$

Modeling Problems: Flows + Cuts

Input: directed G w/ edge capacities $c(e)$
+ $s, t \in V$

Goal: Compute flow $f: E \to \mathbb{R}$ s.t.

① $0 \le f(e) \le c(e)$

② $\forall v \ne s, t,$

$$\sum_u f(u \to v) = \sum_w f(v \to w)$$

Make an LP: Maximize

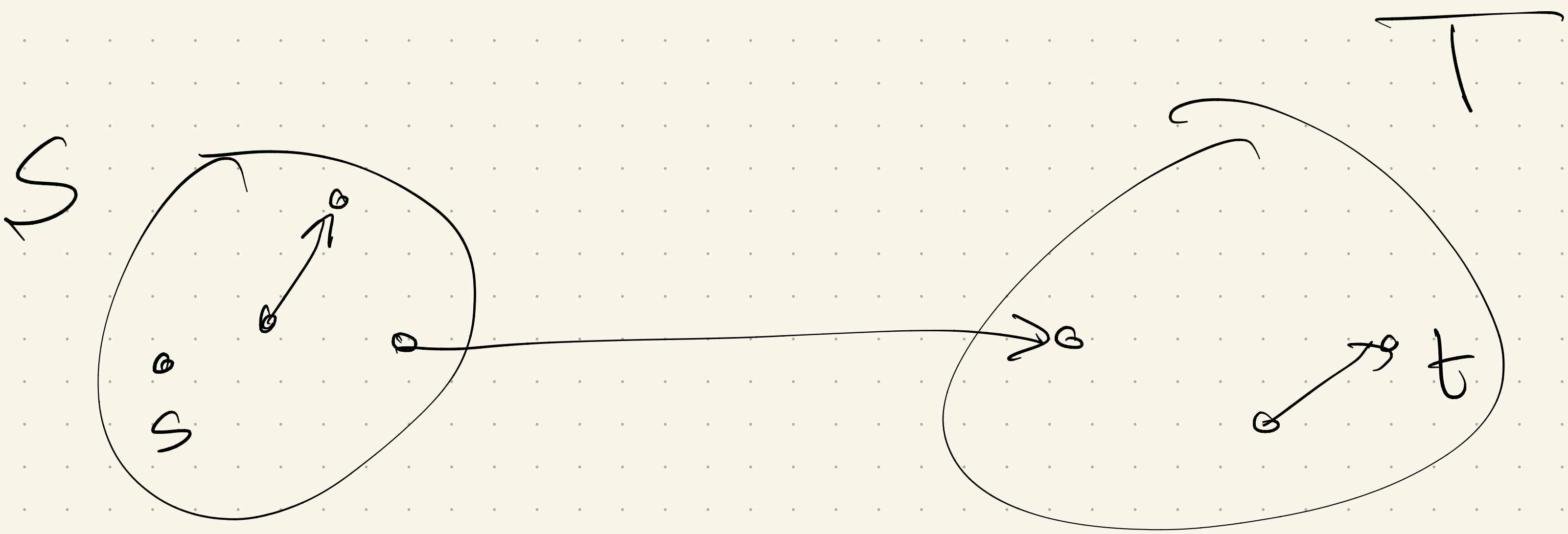
s.t.

Related: Min cuts (S,T)

Use indicator variables:

$$s_v = 0 \text{ or } 1$$

$$x_e = x_{(u\to v)} = 1 \text{ if } u \in S \text{ and } v \in T$$



The LP:

Minimize $\sum_{u\to v} c_{u\to v} \cdot x_{u\to v}$ ← want few edges

s.t.

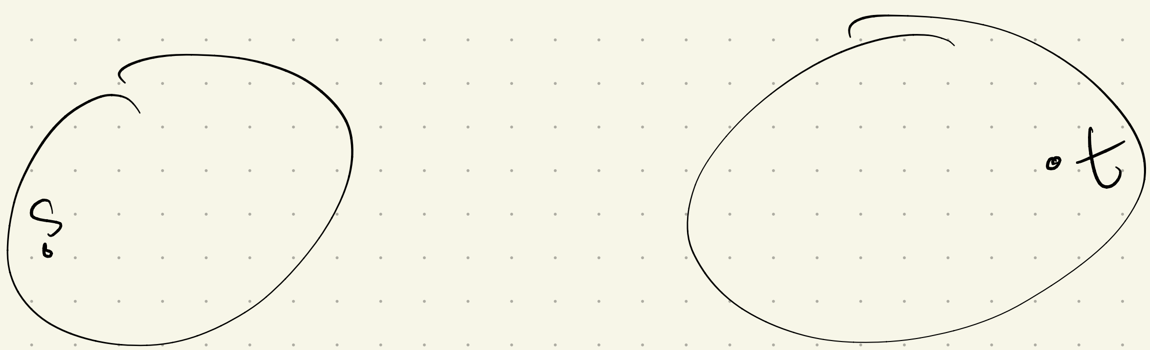
$$x_{u\to v} + s_v - s_u \geq 0 \quad \forall u \to v$$

$$x_{u\to v} \geq 0 \quad \forall u,v$$

$$s_s = 1$$

$$s_t = 0$$

Which are forced?



Note:

For flow/cuts, a solution would yield optimal LP solution.

The reverse is not obvious!

LP might have strange fractional answer which doesn't describe a cut.

It can be shown that this won't happen

↳ but not obvious...

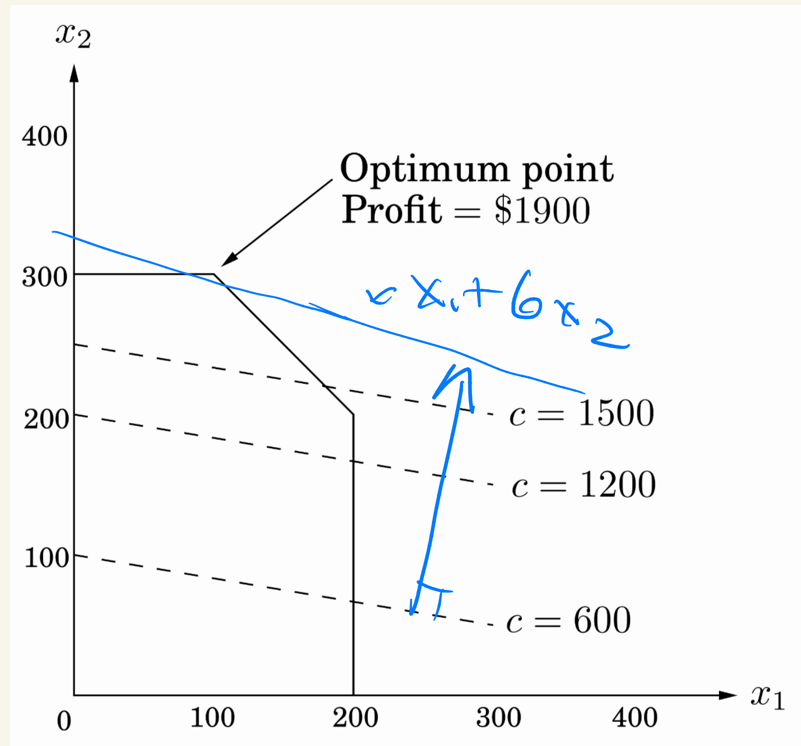
Duality:

Recall our chocolate:

LP: $\max\ x_1 + 6x_2$

s.t.

$$x_1 \le 200$$
$$x_2 \le 300$$
$$x_1 + x_2 \le 400$$
$$x_1, x_2 \ge 0$$



Can we check that this is best?

$$\max \quad x_1 + 6x_2$$

s.t.

$$x_1 \le 200 \qquad (1)$$

$$x_2 \le 300 \qquad (2)$$

$$x_1 + x_2 \le 400$$

$$x_1, x_2 \ge 0$$

Play w/ inequalities:

$(1) + 6 \cdot (2)$:

Interesting!

These 2 inequalities tell us that we couldn't ever beat \$2000.

But recall soln was \$1900 — can we get a better combo?

$$\max \quad x_1 + 6x_2$$

s.t.

$$x_1 \le 200 \qquad (1)$$
$$x_2 \le 300 \qquad (2)$$
$$x_1 + x_2 \le 400 \qquad (3)$$
$$x_1, x_2 \ge 0$$

Play: $0 \cdot (1) + 5 (2) + 1 \cdot (3)$

These multipliers, $(0, 5, 1)$, are a <u>certificate</u> of optimality.

↳ No valid solution can ever beat \$1900

<u>But</u> how do we find these magic values??

In this, we had three "$\leq$" inequalities

↳ so goal is to find the right 3 multipliers: y_1, y_2, and y_3

Let's try to rewrite...

Multiplier		Inequality
y_1	$\times$	$x_1 \leq 200$
y_2	$\times$	$x_2 \leq 300$
y_3	$\times$	$x_1 + x_2 \leq 400$

Result

Rewrite: Make left side look like the original max/min goal, so right will be an upper bound

So here:

$$(y_1+y_3)x_1 + (y_2+y_3)x_2 \leq 200y_1 + 300y_2 + 400y_3$$

Means:

$$x_1 + 6x_2 \leq 200y_1 + 300y_2 + 400y_3$$

if: $\begin{cases} y_1, y_2, y_3 \geq 0 \\ y_1+y_3 \geq 1 \\ y_2+y_3 \geq 6 \end{cases}$

$y_1, y_2, y_3 \geq 0$] b/c if negative, inequalities flip!

$y_1+y_3 \geq 1$, $y_2+y_3 \geq 6$ } b/c original eqn.

Any y_i's would give an upper bound!

We want the best one

↳ ie minimize another LP!

Duality:

$$\begin{aligned} \max \quad & x_1 + 6x_2 \\ \text{s.t.} \quad & x_1 \le 200 \\ & x_2 \le 300 \\ & x_1 + x_2 \le 400 \\ & x_1, x_2 \ge 0 \end{aligned} \quad \overset{\text{dual}}{\Longleftrightarrow} \quad \begin{aligned} \min \quad & 200y_1 + 300y_2 + 400y_3 \\ \text{s.t.} \quad & y_1 + y_3 \ge 1 \\ & y_2 + y_3 \ge 6 \\ & y_1, y_2, y_3 \ge 0 \end{aligned}$$

Any solution to bottom is upper bound to top LP.

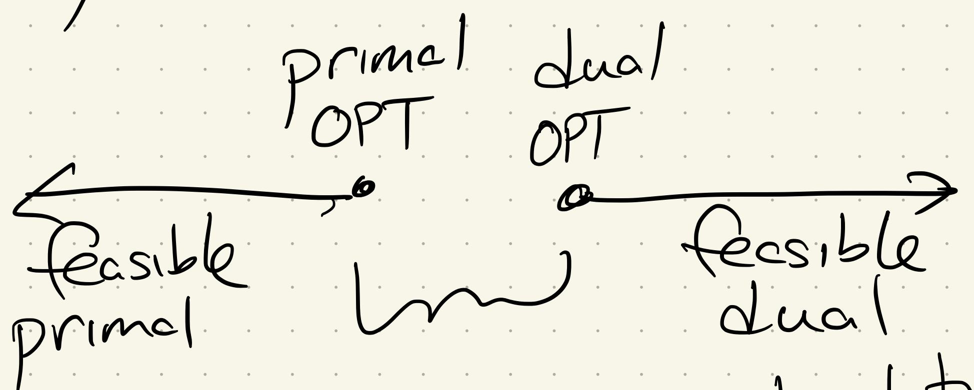
$\Longrightarrow$ If we can find primal/duals that are equal, both are OPT

Here, 1900: primal $(x_1, x_2) = (100, 300)$

Dual: $(y_1, y_2, y_3) = (0, 5, 1)$

This is just like max flow / min cut duality, in a way.

Works for <u>any</u> LP:



this gap - the <u>duality gap</u> - is $= 0$.

In general:

Primal LP	Dual LP
$\max \; c^T x$	$\min \; y^T b$
s.t.	s.t.
$A\vec{x} \leq \vec{b}$	$y^T A \geq c^T$
$\vec{x} \geq 0$	$y \geq 0$
$\vec{c}, \vec{x}, \vec{b}$ vectors	$\vec{c}, \vec{b}, \vec{y}$, vectors

Limits of LP:

Many things are not LPs!

Ex: Integer solutions

Ex: Quadratic or more complex constraints

Often other approaches could give a better runtime!

Ex: Flows & cuts!

Orlin: $O(VE)$ via a combinatorial approach

$\boxed{\begin{array}{l}\underline{\text{LP}}\text{:} \\ E \text{ variables,} \\ V+E \text{ equations}\end{array}}$ $\Rightarrow$

- LP w/ simplex alg. is exponential
- with "better" algorithm matrix multiply time (so $\sim E^3$)

The algorithm: Simplex (Dantzig 1947)

Assumes canonical form:

$$\begin{aligned} &\text{maximize} \sum_{j=1}^{d} c_j x_j \\ &\text{subject to} \sum_{j=1}^{d} a_{ij} x_j \le b_i \quad \text{for each } i = 1 \mathinner{..} n \\ &\qquad\qquad x_j \ge 0 \quad \text{for each } j = 1 \mathinner{..} d \end{aligned}$$

So:

- no min
- only $\le$
- + ≥ 0 for all variables
- fast in practice, but exponential in worst case

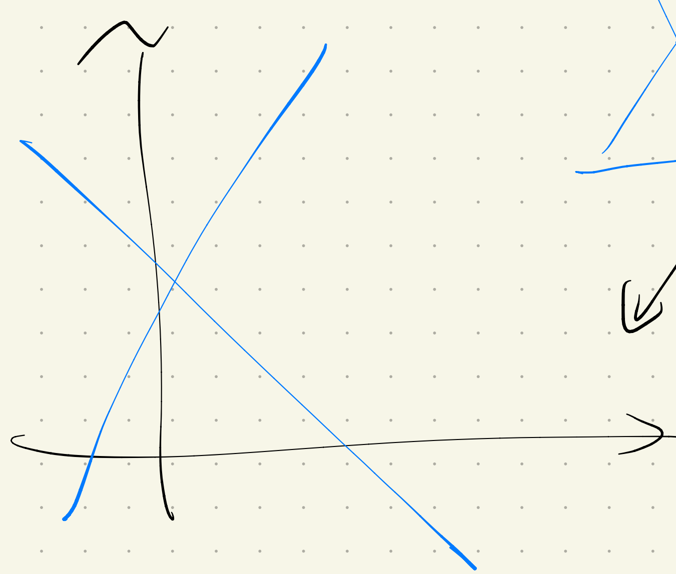
$\boxed{\text{Klee-Minty, 1973: some feasible polytopes have } O(n^{\lfloor d/2 \rfloor}) \text{ vertices}}$

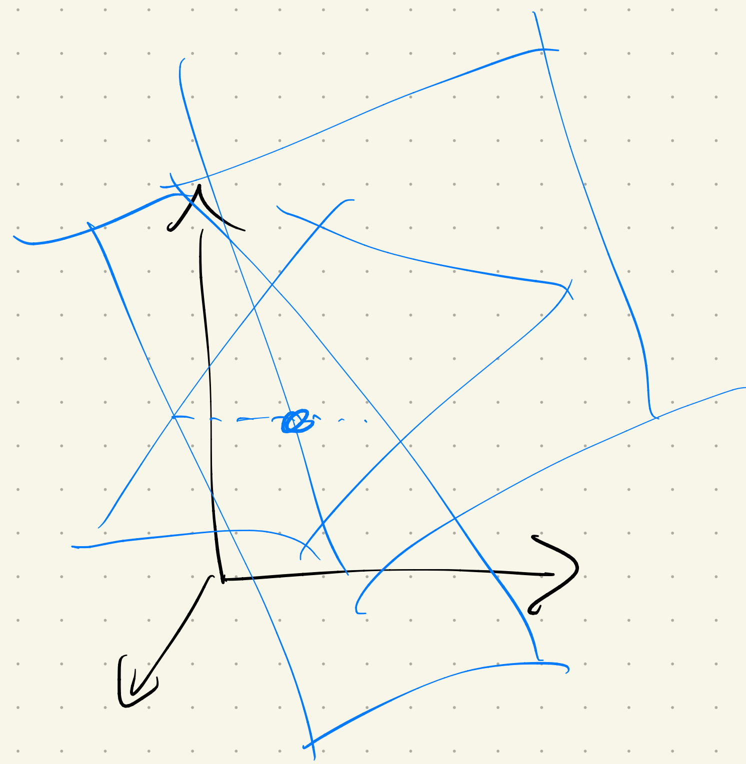
Algorithm (simplex):

Take any vertex v in feasible region
while some neighbor v' is better
$v \leftarrow v'$

Details lacking!

Step 1: find a vertex
take d hyperplanes!





Any d hyperplanes give a vertex:
Loop through m-d others!

Either feasible for each

Or Not

↳ use this new hyperplane

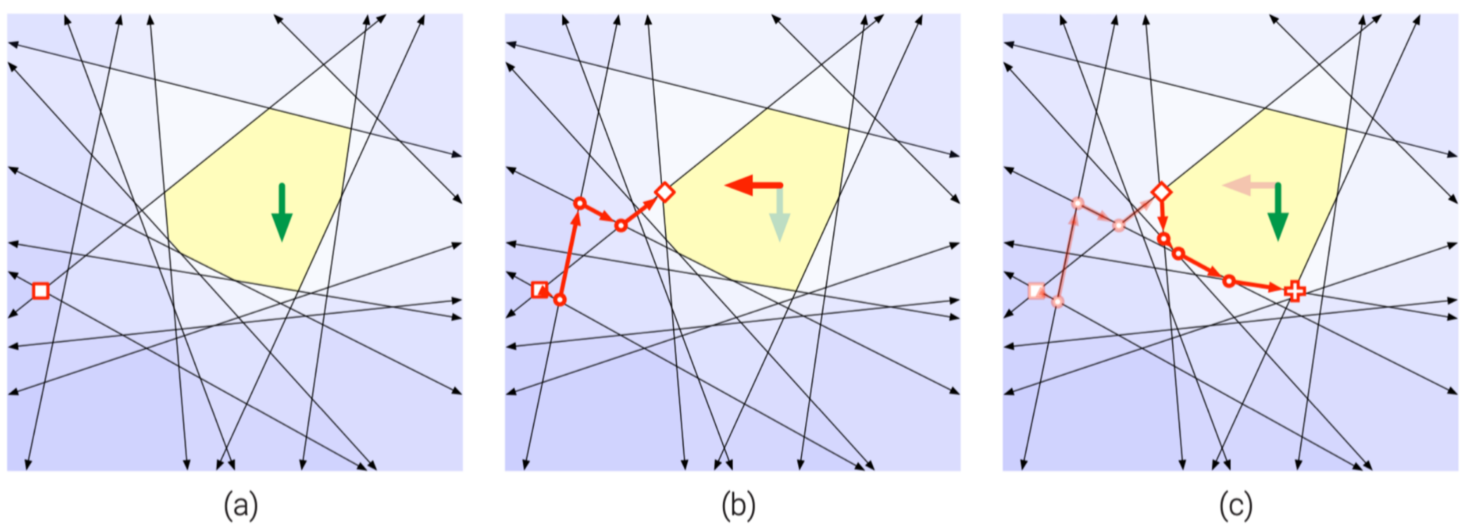


Figure I.2. Primal simplex with dual initialization: (a) Choose any basis. (b) Rotate the objective to make the basis locally optimal, and pivot "up" to a feasible basis. (c) Pivot down to the optimum basis for the original objective.

How to pick where to move?

No ideal way!

Many proposed, but for almost every one, there is <u>some</u> input polyhedron that needs an <u>exponential</u> number of pivots.

Ellipsoid algorithm, Khachiyan 1979
- (weakly) polynomial time
  - ↳ dependent of precision
- high level idea: compute smaller & smaller ellipses which contain solution

Interior point Methods, Karmarkar 1984

- Move through polytope's interior:
  - Still weekly polynomial
    - but practical