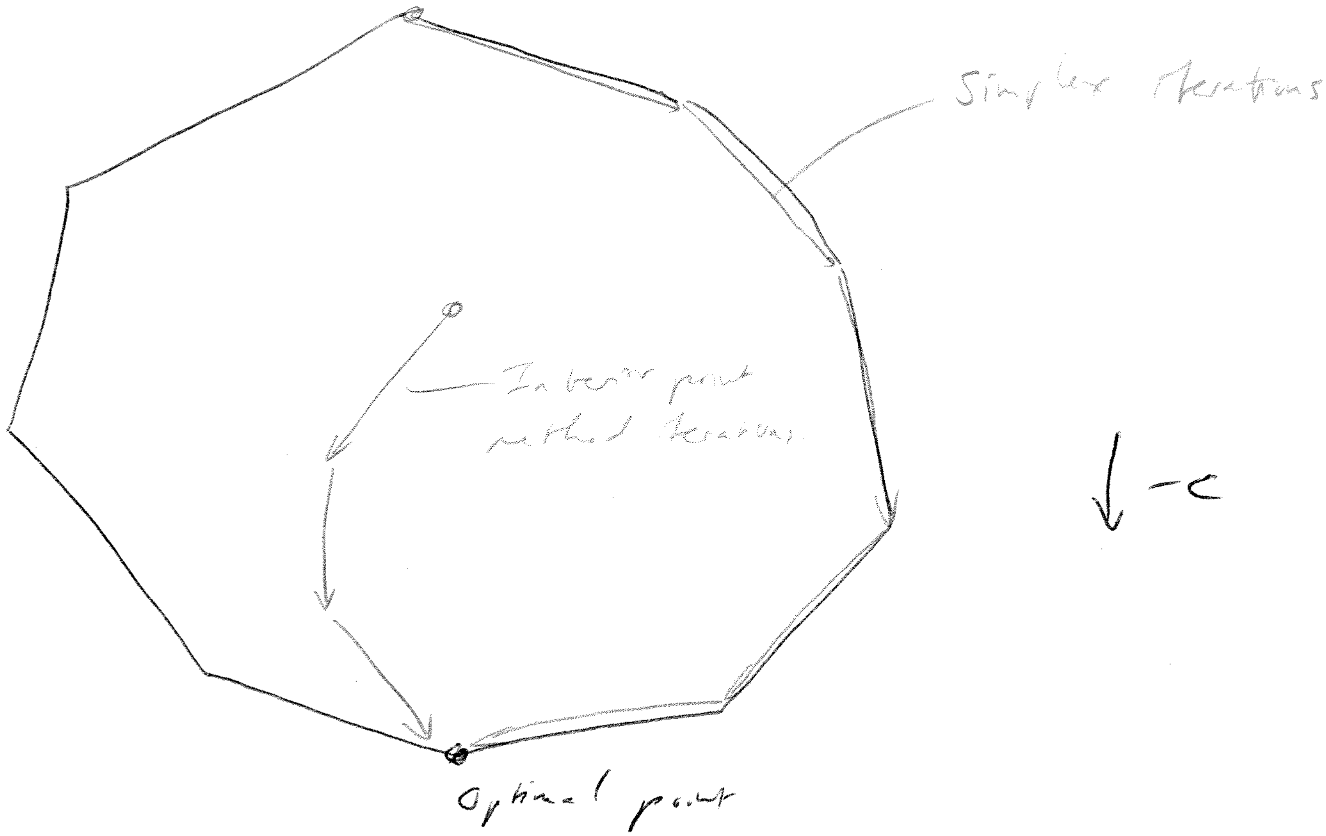
INTERIOR POINT METHODS

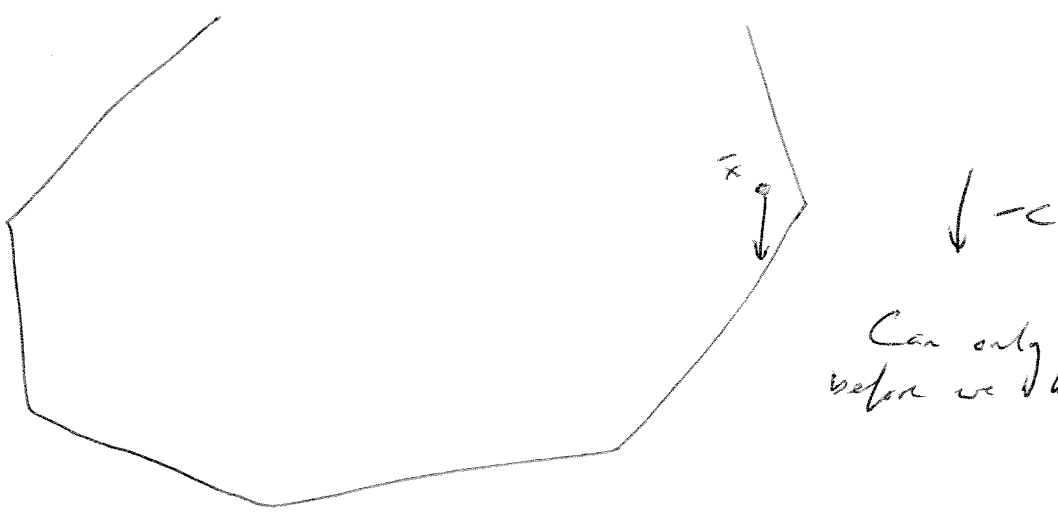
~~Simplex~~



rimal affine scaling variant of Karmarkar's Algorithm.

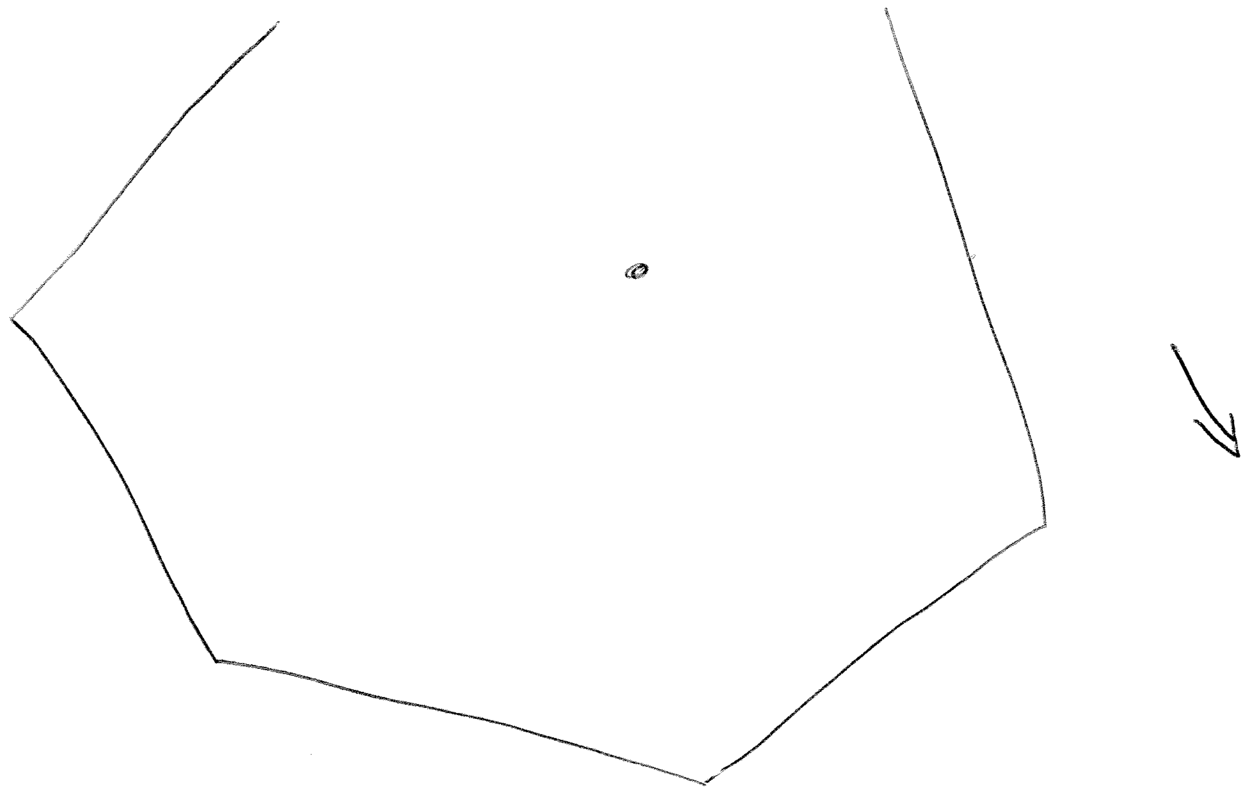
Consider standard form problem
$$\min c^T x \quad Ax = b \quad x \geq 0$$

Assume we know a feasible, strictly positive point $\bar{x}$.



Can only take a short step before we hit the boundary.

So rescale so our current point is near the center:



Now can take long steps.

Walls of polytope are given by nonnegativity constraints $x \geq 0$.

So rescaling corresponds to a change of units.

Rescale so current point is at $\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$.

Then can move distance 1 in any direction before hitting boundary.

OK = 12]

Primal Affine Scaling Method.

Details:

Let x^k be our current point

Let $D^k = \begin{bmatrix} x_1^k & & 0 \\ & x_2^k & \\ & & \ddots \\ 0 & & & x_n^k \end{bmatrix}$, a diagonal $n \times n$ matrix.

Then $(D^k)^{-1} = \begin{bmatrix} \frac{1}{x_1^k} & & \\ & \ddots & \\ & & \frac{1}{x_n^k} \end{bmatrix}$, so $(D^k)^{-1} x^k = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} =: \mathbf{e}$

Now, $Ax = b \iff AD^k(D^k)^{-1}x = b$,

so we replace $\begin{array}{l} \min c^T x \\ Ax = b \quad (P) \\ x \geq 0 \end{array}$ by $\begin{array}{l} \min (D^k c)^T z \\ \text{s.t. } (AD^k) z = b \\ \qquad z \geq 0. \end{array} \quad (P^k)$

Problems are equivalent:

if $\bar{x}$ is feasible in (P) then $\bar{z} = (D^k)^{-1}\bar{x}$ is feasible in (P^k)

If $\tilde{z}$ is feasible in (P^k) then $\tilde{x} = (D^k)\tilde{z}$ is feasible in (P).

Also, $z = e$ is feasible in (P^k), and $D^k e = x^k$.

Also, if $\tilde{z}$ is feasible for (P^k) then

$$(D^k c)^T \tilde{z} = \cancel{(D^k)^T (D^k} = c^T (D^k \tilde{z}) = c^T \tilde{x},$$

ie corresponding points in (P) and (P^k) have the same value.

Write (P^k) as $\begin{array}{l} \min (c^k)^T z \\ \text{s.t. } A^k z = b \quad (P^k) \\ \qquad z \geq 0. \end{array}$

Concentrate on (P^k):

To get maximum decrease in objective function for (P^k), move in the direction $-c^k$.

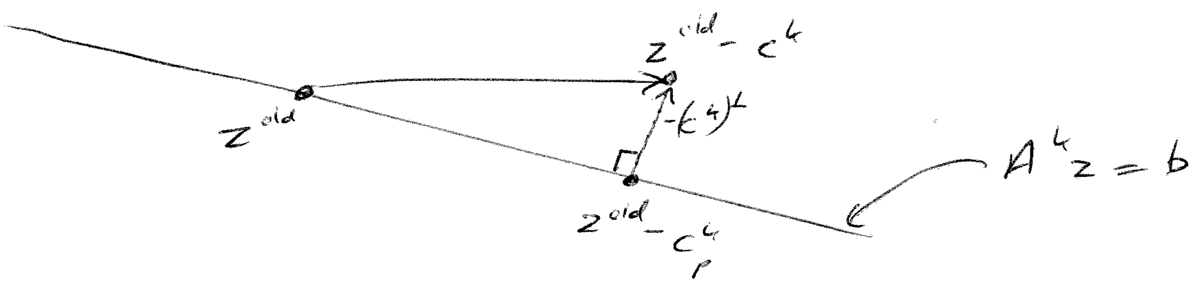
So $z^{new} = z^{old} - \beta c^k$ for some steplength β.

But this is probably infeasible, because we need $Az = b$

$$A^k z^{new} = A^k z^{old} - \beta A^k c^k = b - \beta A^k c^k$$

So we need a direction d with $Ad = 0$.

So: project c^k onto the nullspace of A^k to get direction c^k_p:



So update $z^{new} \leftarrow z^{old} - \alpha c^k_p$ for some α.

Choose α so that $z^{new} \geq 0$.

Then $x^{k+1} = D^k z^{new}$

Example of Primal Affine Scaling Algorithm

$$
\begin{aligned}
\min \quad & -x_1 - x_2 \\
\text{s.t.} \quad & x_1 + 2x_2 + x_3 = 2 \\
& 2x_1 + x_2 + x_4 = 2 \\
& x_i \geq 0.
\end{aligned}
$$

Initialize: $k=0, \quad x^0 = (.3 \quad .4 \quad .9 \quad 1) \quad c^T x^0 = -0.7$

$$D^0 = \begin{bmatrix} .3 & 0 & 0 & 0 \\ 0 & .4 & 0 & 0 \\ 0 & 0 & .9 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A^0 = AD^0 = \begin{bmatrix} .3 & .8 & .9 & 0 \\ .6 & .4 & 0 & 1 \end{bmatrix}$$

$$c^0 = \begin{bmatrix} -.3 & -.4 & 0 & 0 \end{bmatrix}^T$$

First iteration

Calculate $A^0 c^0 = \begin{bmatrix} .3 & .8 & .9 & 0 \\ .6 & .4 & 0 & 1 \end{bmatrix} \begin{bmatrix} -.3 \\ -.4 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -.41 \\ -.34 \end{bmatrix}$

Calculate $A^0 A^{0T} = \begin{bmatrix} .3 & .8 & .9 & 0 \\ .6 & .4 & 0 & 1 \end{bmatrix} \begin{bmatrix} .3 & .6 \\ .8 & .4 \\ .9 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1.54 & .5 \\ .5 & 1.52 \end{bmatrix}$

Calculate $(A^0 A^{0T})^{-1} = \dfrac{1}{2.0908} \begin{bmatrix} 1.52 & -.5 \\ -.5 & 1.54 \end{bmatrix}$

Calculate $v^0 = (A^0 A^{0T})^{-1} A^0 c^0 = \dfrac{-1}{2.0908} \begin{bmatrix} .4532 \\ .3186 \end{bmatrix}$

Calculate $A^{0T} v^0 = \dfrac{1}{2.0908} \begin{bmatrix} .32712 \\ .49 \\ .40788 \\ .3186 \end{bmatrix}$

Calculate $d^0 = -c^0 + A^{0T} v^0 = \begin{bmatrix} .1435432 \\ .16564 \\ -.1950832 \\ -.1523818 \end{bmatrix}$

Calculate steplength $\alpha^0 = .9/\max\{-d_i^0 : d_i^0 < 0\}$

$$= .9/.1950832 = 4.6134162$$

Calculate $x^1 = x^0 + \alpha^0 D^0 d^0 = \begin{bmatrix} .3 \\ .4 \\ .9 \\ 1 \end{bmatrix} + 4.6134162 \begin{bmatrix} .0430629 \\ .066256 \\ -.1755748 \\ -.1523818 \end{bmatrix}$

$$= \begin{bmatrix} .498667 \\ .7056664 \\ .090000 \\ .2969994 \end{bmatrix} \approx \begin{bmatrix} .5 \\ .7 \\ .1 \\ .3 \end{bmatrix}$$

So $x^1 = (.5 \;\; .7 \;\; .1 \;\; .3)$, $k = 1$, $c^T x^1 = -1.2$

Second iteration

$$D^1 = \begin{bmatrix} .5 & 0 & 0 & 0 \\ 0 & .7 & 0 & 0 \\ 0 & 0 & .1 & 0 \\ 0 & 0 & 0 & .3 \end{bmatrix} \qquad A^1 = AD^1 = \begin{bmatrix} .5 & 1.4 & .1 & 0 \\ 1 & .7 & 0 & .3 \end{bmatrix}$$

$$c^1 = [-.5 \;\; -.7 \;\; 0 \;\; 0]^T$$

Calculate $A^1 c^1 = \begin{bmatrix} .5 & 1.4 & .1 & 0 \\ 1 & .7 & 0 & .3 \end{bmatrix} \begin{bmatrix} -.5 \\ -.7 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -1.23 \\ -0.99 \end{bmatrix}$

Calculate $A^1 A^{1T} = \begin{bmatrix} 2.22 & 1.48 \\ 1.48 & 1.58 \end{bmatrix}$

Calculate $(A^1 A^{1T})^{-1} = \dfrac{1}{1.3172} \begin{bmatrix} 1.58 & -1.48 \\ -1.48 & 2.22 \end{bmatrix}$

Calculate $v^1 = (A^1 A^{1T})^{-1} A^1 c^1 = \dfrac{-1}{1.3172} \begin{bmatrix} .4782 \\ .3774 \end{bmatrix}$

Calculate $A^{1T} v^1 = \dfrac{1}{1.3172} \begin{bmatrix} .6165 \\ .93366 \\ .04782 \\ .11322 \end{bmatrix}$

Calculate $d^1 = -c^1 + A^{1T} v^1 = \begin{bmatrix} .0319618 \\ -.0088217 \\ -.0363042 \\ -.085955 \end{bmatrix}$

Calculate step length $\alpha^1 = .9/\max\{-d_i^1 : d_i^1 < 0\} = 10.470595$

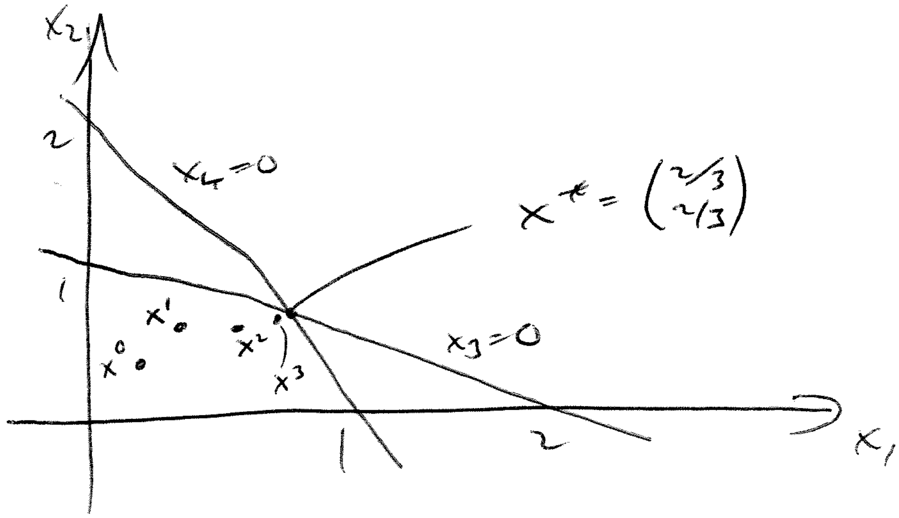
Calculate $x^2 = x^1 + \alpha^1 D^1 d^1 = \begin{bmatrix} .5 \\ .7 \\ .1 \\ .3 \end{bmatrix} + 10.470595 \begin{bmatrix} .0159809 \\ -.0061751 \\ -.00363042 \\ -.0257865 \end{bmatrix}$

$= \begin{bmatrix} .6673295 \\ .6353431 \\ .0619876 \\ .03 \end{bmatrix} \qquad c^T x^2 = -1.302$

We are getting close to the optimal solution $x^* = \begin{bmatrix} 2/3 \\ 2/3 \\ 0 \\ 0 \end{bmatrix}$.

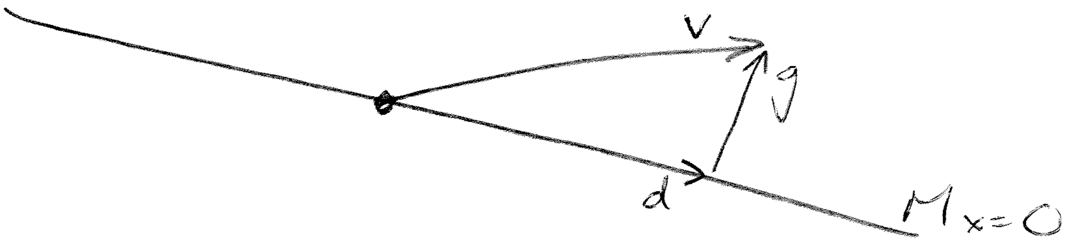
<u>Third iteration</u>

Get $x^3 = \begin{bmatrix} .663 \\ .666 \\ .005 \\ .008 \end{bmatrix} \qquad c^T x^3 = -1.329$


x_2
2
1
$x_4 = 0$
$x^* = \begin{pmatrix} 2/3 \\ 2/3 \end{pmatrix}$
x^0
x^1
x^2
x^3
$x_3 = 0$
1
2
x_1

Calculating projections.

Say we want to calculate the projection d of a vector v onto the nullspace of a matrix M. Assume M is $m\times n$, $n \geq m$, $\mathrm{rank}(M)=m$.



So $v = d + g$ and $Md = 0$, $g = M^T u$ for some u, since null space & row space orthogonal.

Then $Mv = Md + Mg = Mg = MM^T u$

and thus $u = (MM^T)^{-1} Mv$,

so $g = M^T u = M^T (MM^T)^{-1} Mv$

and $d = v - g = v - M^T(MM^T)^{-1}Mv$

$$= \underbrace{\left(I - M^T(MM^T)^{-1}M\right)}_{P_M} v$$

More centering

(Postpone this to discussion of primal-dual methods.)

Sometimes algorithm still gets stuck in corners.

Simple remedies:

(i) So move in direction $D^k P_{A^k} e$ — tries to increase all x_i.

Ie, $x^{k+1} = x^k \cancel{\mp} \beta^k D^k P_{A^k} c^k + \gamma^k D^k P_{A^k} e$.

With appropriate choices, can show convergence in a ~~polynomial~~ number of iterations which is polynomial in m, n, and the ~~[illegible]~~ log(largest entry in A, b, c).

(ii) Only move $\frac{2}{3}$ of way to boundary. Get ~~poly algo.~~ convergent algo.

Can use the calculation of $D^k P_{A^k} e$ to give a more sophisticated dual update than v^k.

An iteration of algorithm

(0) x^k = current iterate, $D^k = \begin{bmatrix} x^k_1 & & 0 \\ & \ddots & \\ 0 & & x^k_n \end{bmatrix}$.

(1) Let $A^k = AD^k$, $c^k = D^k c$

(2) Calculate $v^k = (A^k A^{kT})^{-1} A^k c^k$

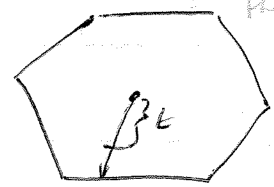
(3) Calculate $d^k = -c^k + A^{kT} v^k = -P_{A^k} c^k$.

Break into two parts: min ratio to find $\max x_i$, then $\beta = t/\max$

(4) Calculate steplength β^k:

$$\beta^k = t \,/\, \max\{-d^k_i : d^k_i < 0\}$$

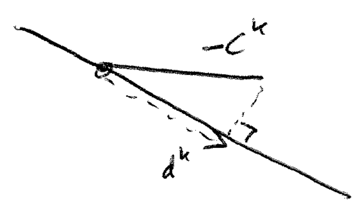
(t = proportion of step to boundary)



(5) Update $x^{k+1} = x^k + \beta^k X^k d^k$.

Monotonicity

$$\begin{aligned}
c^T x^k - c^T x^{k+1} &= c^T x^k - c^T(x^k + \beta^k X^k d^k) \\
&= -\beta^k c^T X^k d^k \\
&= \beta^k c^T X^k P_{A^k} c^k \\
&= \beta^k c^{kT} P_{A^k} c^k \\
&= \beta^k \| P_{A^k} c^k \|_2^2. \\
&> 0
\end{aligned}$$



Dual solution

It can be shown that: (i) $D^k d^k = 0 \Longleftrightarrow x^k$ is a vertex (Complementary slackness)

(ii) $d^k \leq 0$ at optimal vertex
Optimal vertex is only vertex where $d^k \leq 0$.

Try $v^k = (A^k A^{kT})^{-1} A^k c^k$ as dual solution:

Dual feasibility?

$$A^T (A^k A^{kT})^{-1} A^k c^k \leq c^k \ ? \quad \text{Yes, if } d^k \leq 0$$

It can be shown that $b^T v^k = c^T x^k$ at optimal soln.

Termination

It can be shown that the algorithm converges to the optimal solution, provided $\epsilon \leq 2/3$.

Possible termination criteria:

(i) Is $c - A^T v^k \geq -\varepsilon e$? Duality

(ii) Is $\| d \|$ small ?

(iii) Is $\dfrac{c^T x^k - c^T x^{k+1}}{\max\{1, |c^T x^k|\}}$ small ?

} Not making much progress.

(iv) Is $D^k (c - A^T v^k)$ small ? Complementary slackness

LOG BARRIER METHODS FOR INTERIOR POINT SEARCH

Barrier objective functions

Standard form LP: $\min c^T x$
$\quad$ s.t. $Ax = b$
$\quad\quad x \geq 0$.

Modify objective to discourage approaching boundary: $\min c^T x - \mu \sum_{j=1}^{n} \ln x_j$

Note: as $x_j \to 0$, $\ln x_j \to -\infty$, so obj $\to +\infty$.
Use here, $\mu > 0$.

Eg: $\min \; -x_1 - x_2$
s.t. $x_1 + 2x_2 + x_3 = 2$
$2x_1 + x_2 + x_4 = 2$
$x_i \geq 0$.

Barrier objective function: $-x_1 - x_2 - \mu\ln x_1 - \mu\ln x_2 - \mu\ln x_3 - \mu\ln x_4$.

At $x = (.3, .4, .9, 1)$: $-x_1 - x_2 = -0.7$
If $\mu = 2$: $\quad -x_1 - x_2 - \mu\sum_{j=1}^{4}\ln x_j \approx 3.751\ldots$
If $\mu = 0.01$: $\quad -x_1 - x_2 - \mu\sum_{j=1}^{4}\ln x_j \approx -.677\ldots$

At $x = (.66, .66, .02, .02)$: $-x_1 - x_2 = -1.32$
If $\mu = 2$: $\quad -x_1 - x_2 - \mu\sum \ln x_j \approx 6.182\ldots$
If $\mu = 0.01$: $\quad -x_1 - x_2 - \mu\sum \ln x_j \approx -1.282\ldots$

If $\mu = 2$, $(.3, .4, .9, 1)$ looks more attractive.
If $\mu = .01$, other point is more attractive.
The barrier term penalizes approaching the boundary. Penalty is less for smaller values of μ

So solve: $\min \; \cancel{f(x)}\, f(x) := c^T x - \mu \sum \ln x_j$
s.t. $Ax = b$.

Can ignore nonnegativity constraints because barrier term will keep us away from the boundary

Use second order Taylor approximation to objective:

$$f(x^k + \lambda \Delta x) \approx f(x^k) + \lambda \Delta x^T \nabla f(x^k) + \frac{\lambda^2}{2} \Delta x^T \nabla^2 f(x^k) \Delta x$$

where $\nabla f(x^k)$ = gradient of f at x^k, ~~so~~ $\cancel{(\nabla f(x^k))_i = \frac{\partial f}{\partial x_k}}$

$$\text{so } (\nabla f(x^k))_i = \frac{\partial f}{\partial x_i^k}(x^k) = c_i - \frac{\mu}{x_i^k}$$

and $\nabla^2 f(x^k)$ = matrix of second derivatives of f at x^k

$$\text{so } (\nabla^2 f(x^k))_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}(x^k) = \frac{\mu}{x_i^k x_j^k}$$

Try to minimize this approximation ~~as~~ ~~Az~~ with the restriction that $A \Delta x = 0$.

Get: $\Delta x = -\frac{1}{\mu} X^k P_{AX^k}(X^k c - \underbrace{\mu e}_{})$ this is only difference from affine method. Pushes iterates towards center.

where $X^k = \begin{bmatrix} x_1^k & & \\ & \ddots & \\ & & x_n^k \end{bmatrix}$, so $X^k c - \mu e = \begin{bmatrix} x_1^k c_1 - \mu \\ \vdots \\ x_n^k c_n - \mu \end{bmatrix}$

and P_{AX^k} denotes projection onto null-space of AX^k.
Called NEWTON STEP (cf NEWTON-RAPHSON METHOD)

Eg: $\min\ -x_1 - x_2 - 0.5\sum_{i=1}^{4} \ln x_i$ with $x^k = (.3, .4, .9, 1)$

s.t. $x_1 + 2x_2 + x_3 = 2$

$2x_1 + x_2 + x_4 = 2$

(and so $\mu = 0.5$):

$$X^k c - \mu e = \begin{bmatrix} -.3 \\ -.4 \\ 0 \\ 0 \end{bmatrix} - 0.5 \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -.8 \\ -.9 \\ -.5 \\ -.5 \end{bmatrix}$$

$$P_{AX^k} = I - X^k A^T (AX^{k^2}A^T)^{-1} AX^k$$

$$AX^k = \begin{bmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} .3 & & & \\ & .4 & & \\ & & .9 & \\ & & & 1 \end{bmatrix} = \begin{bmatrix} .3 & .8 & .9 & 0 \\ .6 & .4 & 0 & 1 \end{bmatrix}$$

$$AX^{k^2}A^T = \begin{bmatrix} .3 & .8 & .9 & 0 \\ .6 & .4 & 0 & 1 \end{bmatrix} \begin{bmatrix} .3 & .6 \\ .8 & .4 \\ .9 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1.54 & 0.5 \\ 0.5 & 1.52 \end{bmatrix}$$

$$(AX^{k^2}A^T)^{-1} = \frac{1}{2.0908} \begin{bmatrix} 1.52 & -0.5 \\ -0.5 & 1.54 \end{bmatrix}$$

Thus
$$\Delta x = \begin{bmatrix} 0.119 \\ 0.061 \\ -0.241 \\ -0.300 \end{bmatrix}$$

Check: $A\Delta x =$

$$\Delta x = -\frac{1}{\mu} X^k (X^k c - \mu e) + \frac{1}{\mu} X^{k^2} A^k (AX^{k^2}A^T)^{-1} AX^k (X^k c - \mu e)$$

$$= -2 \begin{bmatrix} .3 & & & \\ & .4 & & \\ & & .9 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} -.8 \\ -.9 \\ -.5 \\ -.5 \end{bmatrix} + \frac{2}{2.0908} \begin{bmatrix} .3 & & & \\ & .4 & & \\ & & .9 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} .3 & .6 \\ .8 & .4 \\ .9 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1.52 & -0.5 \\ -0.5 & 1.54 \end{bmatrix} \begin{bmatrix} .3 & .8 & .9 & 0 \\ .6 & .4 & 0 & 1 \end{bmatrix} \begin{bmatrix} -.8 \\ -.9 \\ -.5 \\ -.5 \end{bmatrix} =$$

Primal-Dual interior point barrier search.

Used in commercial solvers eg CPLEX.

$$\min \; c^T x - \mu \sum \ln x_j \qquad P(\mu)$$
$$\text{s.t.} \quad Ax = b.$$

Theorem A vector $x > 0$ is optimal in $P(\mu)$ if and only if there exist vector y and z satisfying:

$$Ax = b$$
$$A^T y + z = c$$
$$x_i z_i = \mu, \quad i = 1, \dots, n$$

Ie, primal feasibility, dual feasibility, and μ-complementary slackness

Note: $x_i > 0$, $\mu > 0$ forces $z_i > 0$.
Recall: Duality gap $= \sum x_i z_i = z^T x$

Primal-dual move directions

Use a different scaling matrix:

$$D = \begin{bmatrix} \sqrt{x_1/z_1} & & & \\ & \sqrt{x_2/z_2} & & \\ & & \ddots & \\ & & & \sqrt{x_n/z_n} \end{bmatrix}$$

Note: If $x_i z_i = \mu$ then $\frac{1}{z_i} = \frac{x_i}{\mu}$, so $\sqrt{\frac{x_i}{z_i}} = \sqrt{\frac{x_i^2}{\mu}} = \frac{x_i}{\sqrt{\mu}}$, ie like our previous diagonal matrix

Get directions for both x and also y and z:

$$\Delta x = \underbrace{D^k}_{\text{scale back}} \underbrace{P_{AD^k}}_{\text{project}} \underbrace{D^k}_{\text{scale}} \Big(\underbrace{\mu X^{-1} e - z}_{\text{looks like -(gradient) of } z^T x - \mu \sum_{i=1}^{n} \ln x_i} \Big)$$

$$\Delta y = -(AD^{k^2}A^T)^{-1} AD^{k^2} (\mu X^{-1} e - z)$$

$$\Delta z = -A^T \Delta y, \text{ so } A^T \Delta y + \Delta z = 0, \text{ so keep } A^T y + z = c.$$

(Note that $\Delta x = D^k (I - D^k A^T (AD^{k^2}A^T)^{-1} AD^k) D^k (\mu X^{-1} e - z)$

$$= D^{k^2}(\mu X^{-1} e - z) - D^{k^2} \Delta z$$

Also, $D^{k^2}(\mu X^{-1} e - z) = X^k (Z^k)^{-1} (\mu X^{k^{-1}} e - z)$

$$= (Z^k)^{-1} (\mu e - X^k z).$$)

Put on handout.

Primal-dual barrier algorithm

1. Find $x^0 > 0$ with $Ax = b$
2. Find y^0, z^0 with $A^T y + z = c$, $z > 0$.
3. Choose barrier parameter $\mu > 0$. Set counter $k = 0$.
4. Outer Loop:
   (a) If $x^T z$ small enough, STOP.

   (b) Inner loop:
   While still not close to solving $P(\mu)$,
   ie, while $x_i z_i \neq \mu$ for at least some components:
   - Calculate directions $\Delta y, \Delta z, \Delta x$
   - ~~Upd~~ Choose primal and dual step lengths α^P, α^D
   - Update
   $$x^{k+1} \leftarrow x^k + \alpha^P \Delta x$$
   $$y^{k+1} \leftarrow y^k + \alpha^D \Delta y$$
   $$z^{k+1} \leftarrow z^k + \alpha^D \Delta z$$
   $$k \leftarrow k+1$$

   (c) Reduce μ

   (d) Return to (a).

EXAMPLE

$$\begin{aligned} \min \quad & -x_1 - x_2 \\ \text{s.t.} \quad & x_1 + 2x_2 + x_3 = 2 \\ & 2x_1 + x_2 + x_4 = 2 \\ & x_i \geq 0. \end{aligned}$$

$$\begin{aligned} \max \quad & 2y_1 + 2y_2 \\ \text{s.t.} \quad & y_1 + 2y_2 + z_1 = -1 \\ & 2y_1 + y_2 + z_2 = -1 \\ & y_1 + z_3 = 0 \\ & y_2 + z_4 = 0 \end{aligned}$$

Initialize with $x = (0.3, 0.4, 0.9, 1)$
$y = (-2, -2)$, $z = (5, 5, 2, 2)$

Then: $x_1 z_1 = 1.5$, $x_2 z_2 = 2$, $x_3 z_3 = 1.8$, $x_4 z_4 = 2$.

So pretty close if take $\boxed{\mu = 2}$.

Reduce μ: $\mu \leftarrow \mu / \underbrace{4}_{\text{dimension of } x.} = \boxed{0.5}$

Calculate $\Delta y = -(AD^{k^2}A^T)^{-1} A D^{k^2} (\mu X^{-1} e - z)$:

$$= -(AD^{k^2}A^T)^{-1} A(\mu Z^{-1} e - Xe): \text{ since } D^2 = XZ^{-1}.$$

$$\mu Z^{-1} e - Xe = 0.5 \begin{bmatrix} \frac{1}{5} & & & \\ & \frac{1}{5} & & \\ & & \frac{1}{2} & \\ & & & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 0.3 \\ 0.4 \\ 0.9 \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{10} \\ \frac{1}{10} \\ \frac{1}{4} \\ \frac{1}{4} \end{bmatrix} - \begin{bmatrix} 0.3 \\ 0.4 \\ 0\text{[illegible]} \\ \text{[illegible]} \end{bmatrix}$$

$$= \begin{bmatrix} -0.2 \\ -0.3 \\ -0.65 \\ -0.75 \end{bmatrix}$$

Now,

$$\Delta x = D\, P_{AD}\, D(\mu X^{-1}e - z)$$

$$= D\, (I - DA^T(AD^2A^T)^{-1}AD)\, D(\mu X^{-1}e - z)$$

$$= D^2(\mu X^{-1}e - z) - D^2A^T(AD^2A^T)^{-1}AD^2(\mu X^{-1}e - z)$$

$$= D^2(\mu X^{-1}e - z) - D^2 \Delta z$$

$$= \begin{bmatrix} -0.2 \\ -0.3 \\ -0.65 \\ -0.75 \end{bmatrix} - \begin{bmatrix} \frac{0.3}{5} & & & \\ & \frac{0.4}{5} & & \\ & & \frac{0.4}{2} & \\ & & & \frac{1}{2} \end{bmatrix} \begin{bmatrix} -3.95 \\ -3.92 \\ -1.30 \\ -1.32 \end{bmatrix}$$

$$= \begin{bmatrix} 0.037 \\ 0.014 \\ -0.065 \\ -0.088 \end{bmatrix}$$

Use minimum ratio tests to choose steplengths:

$$\alpha_D = 0.9 \min\left\{ \frac{z_i}{-\Delta z_i} : \Delta z_i < 0 \right\} = 1.14$$

$$\alpha_P = 0.9 \min\left\{ \frac{x_i}{-\Delta x_i} : \Delta x_i < 0 \right\} = 10.25$$

Then $A(\mu Z^{-1}e - Xe) = \begin{bmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -0.2 \\ -0.3 \\ -0.65 \\ -0.75 \end{bmatrix} = \begin{bmatrix} -1.45 \\ -1.45 \end{bmatrix}$

Also, $AD^2A^T = \begin{bmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{0.3}{5} & & & \\ & \frac{0.4}{5} & & \\ & & \frac{0.9}{2} & \\ & & & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$

$= \begin{bmatrix} 0.83 & 0.28 \\ 0.28 & 0.82 \end{bmatrix}$

Then $(AD^2A^T)^{-1} = \frac{1}{0.6022} \begin{bmatrix} 0.82 & -0.28 \\ -0.28 & 0.83 \end{bmatrix}$

And $\Delta y = -(AD^2A^T)^{-1} A(\mu Z^{-1}e - Xe)$

$= -\frac{1}{0.6022} \begin{bmatrix} 0.82 & -0.28 \\ -0.28 & 0.83 \end{bmatrix} \begin{bmatrix} -1.45 \\ -1.45 \end{bmatrix}$

$= \begin{bmatrix} 1.30 \\ 1.32 \end{bmatrix}$

Then $\Delta z = -A^T \Delta y = \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1.30 \\ 1.32 \end{bmatrix}$

$= \begin{bmatrix} -3.95 \\ -3.92 \\ -1.30 \\ -1.32 \end{bmatrix}$

Update iterates:

$$x^{k+1} = x^k + \alpha^P \Delta x = \begin{bmatrix} 0.3 \\ 0.4 \\ 0.9 \\ 1.0 \end{bmatrix} + 1.025 \begin{bmatrix} 0.037 \\ 0.014 \\ -0.065 \\ -0.088 \end{bmatrix} = \begin{bmatrix} 0.678 \\ 0.543 \\ 0.235 \\ 0.100 \end{bmatrix}$$

$$y^{k+1} = y^k + \alpha^D \Delta y = \begin{bmatrix} -2 \\ -2 \end{bmatrix} + 1.14 \begin{bmatrix} 1.30 \\ 1.32 \end{bmatrix} = \begin{bmatrix} -0.518 \\ -0.491 \end{bmatrix}$$

$$z^{k+1} = z^k + \alpha^D \Delta z = \begin{bmatrix} 5 \\ 5 \\ 2 \\ 2 \end{bmatrix} + 1.14 \begin{bmatrix} -3.95 \\ -3.92 \\ -1.30 \\ -1.32 \end{bmatrix} = \begin{bmatrix} 0.500 \\ 0.527 \\ 0.518 \\ 0.491 \end{bmatrix}$$

Then

$x_1 z_1 = 0.339 \quad x_2 z_2 = 0.287 \quad x_3 z_3 = 0.122 \quad x_4 z_4 = 0.049$

One more iteration gives: (with $\mu = 0.04$)

$$x^{k+2} = \begin{bmatrix} 0.598 \\ 0.689 \\ 0.024 \\ 0.114 \end{bmatrix}$$

$$y^{k+2} = \begin{bmatrix} -0.335 \\ -0.383 \end{bmatrix}$$

$$z^{k+2} = \begin{bmatrix} +0.101 \\ 0.053 \\ 0.335 \\ 0.383 \end{bmatrix}$$

$x_1 z_1 = 0.060 \qquad x_2 z_2 = 0.036 \qquad x_3 z_3 = 0.008 \qquad x_4 z_4 = 0.044$

Another iterate with $\mu = 0.004$ gives

$$x^{k+3} = \begin{bmatrix} 0.662 \\ 0.664 \\ 0.010 \\ 0.011 \end{bmatrix}.$$

Computational Effort

- Each iteration requires a lot of work
- Only need a few iterations: Perhaps only 40 iterations for a problem with ~10000 constraints/variables
- In practice, don't calculate $(AD^2A^T)^{-1}$, but use Gaussian elimination:

  Want to find ~~AD^2A^T~~ $v = (AD^2A^T)^{-1} g$ for some known vector g

  So solve $(AD^2A^T)v = g$ using Gaussian elimination.

- As for revised simplex, need to exploit the fact that many entries of A may be zero.

  Eg, in project, have perhaps 120-150 variables, but each constraint only involves 2-4 of these.

  (3 process, 2 inv wafers, 4 sell, 3 inv devices, 12 assign × 5 or 6 time periods)

- Interior point methods better than simplex for large problems (>1000 variables or constraints), provided A does not contain too many nonzeroes. See sheet.