Convex Optimization

Prof. Nati Srebro

Lecture 13:

Primal-Dual Interior Point Method

Reading: Boyd and Vandenberghe Section 11.7

The Simplex Method

Reading: Nocedal and Wright Sections 13.2,13.3,13.8
Additional details: Sections 13.4—13.7

$$\min_{x\in\mathbb{R}^n} \quad f_0(x)$$
$$s.t. \quad f_i(x) \le 0,\ Ax = b$$

$$L(x,\lambda,\nu) = f_0(x) + \textstyle\sum_i \lambda_i f_i(x) + \langle \nu, Ax - b\rangle$$

$$\min_{x\in\mathbb{R}^n} \quad f_0(x) - \tfrac{1}{t}\textstyle\sum_{i=1}^m \log\big(-f_i(x)\big)$$
$$s.t. \quad Ax = b$$

$$L_t(x,\nu) = f_0(x) - \tfrac{1}{t}\textstyle\sum_i \log\big(-f_i(x)\big) + \langle \nu, Ax - b\rangle$$

Newton iteration of log-barrier method:

1. Use (C) to eliminate $\lambda_i = \frac{-1}{tf_i(x)}$, substitute to get problem in x, ν.

➔ (D) becomes $0 = \nabla_x L\left(x, \frac{-1}{tf_i(x)}, \nu\right) = \nabla_x L_t(x,\nu)$

2. Linearize (D) about $x = x^{(k)} + \Delta x$
3. Solve (P)+($\widetilde{\text{D}}$) for $\Delta x, \nu$, take step in direction Δx (ensuring progress, and (f)+(λ) remain valid)

Relaxed KKT

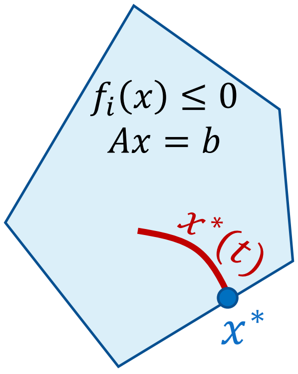
(f) $f_i(x) \le 0$

(λ) $\lambda_i \ge 0$

(P) $Ax = b$

(D) $\nabla_x L(x,\lambda,\nu) = 0$

(C) $\lambda_i \cdot f_i(x) = -{}^1/_t$



$$\min_{x\in\mathbb{R}^n} \quad f_0(x)$$
$$s.t. \quad f_i(x) \le 0,\ Ax = b$$

$L(x,\lambda,\nu) =$
$f_0(x) + \sum_i \lambda_i f_i(x) + \langle \nu, Ax - b\rangle$

$$\min_{x\in\mathbb{R}^n} \quad f_0(x) - \frac{1}{t}\sum_{i=1}^m \log\big(-f_i(x)\big)$$
$$s.t. \quad Ax = b$$

$L_t(x,\nu) =$
$f_0(x) - \frac{1}{t}\sum_i \log\big(-f_i(x)\big) + \langle \nu, Ax - b\rangle$

Newton iteration of log-barrier method:

1. Use (C) to eliminate $\lambda_i = \frac{-1}{t f_i(x)}$, substitute to get problem in x, ν.

➔ (D) becomes $0 = \nabla_x L\left(x, \frac{-1}{t f_i(x)}, \nu\right) = \nabla_x L_t(x,\nu)$

2. Linearize (D) about $x = x^{(k)} + \Delta x$
3. Solve (P)+($\tilde{\text{D}}$) for $\Delta x, \nu$, take step in direction Δx (ensuring progress, and (f)+(λ) remain valid)

Relaxed KKT

(f) $f_i(x) \le 0$

(λ) $\lambda_i \ge 0$

(P) $Ax = b$

(D) $\nabla_x L(x,\lambda,\nu) = 0$

(C) $\lambda_i \cdot f_i(x) = -{}^1/_t$

Iteration of Primal/Dual method: (work on $x^{(k)}, \lambda^{(k)}, \nu^{(k)}$)

1. Linearize (D)+(C) about $x = x^{(k)} + \Delta x$ and $\lambda = \lambda^{(k)} + \Delta x$
2. Solve (P)+($\tilde{\text{D}}$)+($\tilde{\text{C}}$) for $x = x^{(k)} + \Delta x, \lambda = \lambda^{(k)} + \Delta\lambda, \nu = \nu^{(k)} + \Delta\, \nu$ take step in direction $\Delta x, \Delta\lambda\ , \Delta\nu$ (ensuring (f)+(λ) remain valid)

$$\begin{aligned}\min_{x\in\mathbb{R}^n} \quad & f_0(x) \\ s.t. \quad & f_i(x) \le 0 \qquad \lambda \in \mathbb{R}^m \\ & \quad i = 1..m \\ & Ax = b \qquad \nu \in \mathbb{R}^m \\ & \quad A \in \mathbb{R}^{p\times n}\end{aligned}$$

$$r_P(x) = Ax - b \in \mathbb{R}^p$$

$$r_D(x,\lambda,\nu) = \nabla_x L(x,\lambda,\nu) \in \mathbb{R}^n$$

$$r_{C(t)}(x,\lambda) = \left[\lambda_i f_i(x) + \frac{1}{t}\right]_{i=1..m} \in \mathbb{R}^m$$

$$r_t(x,\lambda,\nu) = \left[r_p\ r_D\ r_{C(t)}\right] \in \mathbb{R}^{n+m+p}$$

t-Relaxed KKT

$$f_i(x) \le 0 \qquad \lambda_i \ge 0 \qquad Ax = b \qquad \nabla_x L(x,\lambda,\nu) = 0 \qquad \lambda_i f_i(x) = -{}^1/_t$$

$$r_P(x) = 0 \qquad r_D(x,\lambda,\nu) = 0 \qquad r_{C(t)}(x,\lambda) = 0$$

At each iteration linearize r_t about $x^{(k)}, \lambda^{(k)}$ and solve:

$$r_t\left(x^{(k)} + \Delta x, \lambda^{(k)} + \Delta\lambda, \nu^{(k)} + \Delta\nu\right) \approx$$
$$r\left(x^{(k)}, \lambda^{(k)}, \nu^{(k)} + \Delta\nu\right) + \nabla_x r\left(x^{(k)}, \lambda^{(k)}, \nu^{(k)}\right)^\top \Delta x + \nabla_\lambda r\left(x^{(k)}, \lambda^{(k)}, \nu^{(k)}\right)^\top \Delta\lambda = 0$$

In matrix form:

$$\begin{bmatrix} \nabla^2 f\left(x^{(k)}\right) + \sum \lambda_i^{(k)} \nabla^2 f_i\left(x^{(k)}\right) & \left(\nabla f_i\left(x^{(k)}\right)\right)_i & A^\top \\ \left(\lambda_i^{(k)} \nabla f_i\left(x^{(k)}\right)^\top\right)_i & diag\left(f_i\left(x^{(k)}\right)\right) & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\lambda \\ \Delta\nu \end{bmatrix} = -\begin{bmatrix} r_D\left(x^{(k)}, \lambda^{(k)}, \nu^{(k)}\right) \\ r_{C(t)}\left(x^{(k)}, \lambda^{(k)}\right) \\ r_P\left(x^{(k)}\right) \end{bmatrix}$$

Primal-Dual Interior Point Method

Init: $x^{(0)}, \lambda^{(0)}, \nu^{(0)}$ s.t. $f_i\left(x^{(0)}\right) < 0$, $f_0\left(x^{(0)}\right) < \infty, \lambda^{(0)} > 0$

Iterate:

$t^{(k)} =$...

Solve linearized $t^{(k)}$-relaxed KKT:

$$\begin{bmatrix} \nabla^2 f\left(x^{(k)}\right) + \sum \lambda_i^{(k)} \nabla^2 f_i\left(x^{(k)}\right) & \left(\nabla f_i\left(x^{(k)}\right)\right)_i & A^\top \\ \left(\lambda_i^{(k)} \nabla f_i\left(x^{(k)}\right)^\top\right)_i & diag\left(f_i\left(x^{(k)}\right)\right) & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta \nu \end{bmatrix} = -\begin{bmatrix} r_d\left(x^{(k)}, \lambda^{(k)}, \nu^{(k)}\right) \\ r_{c\left(t^{(k)}\right)}\left(x^{(k)}, \lambda^{(k)}\right) \\ r_p\left(x^{(k)}\right) \end{bmatrix}$$

Set stepsize s by backtracking linesearch on $\left\|r_{t^{(k)}}\right\|$,

ensuring $f_i(x) < 0$ and $\lambda_i > 0$

$\left(x^{(k+1)}, \lambda^{(k+1)}, \nu^{(k+1)}\right) \leftarrow \left(x^{(k+1)}, \lambda^{(k+1)}, \nu^{(k+1)}\right) + s(\Delta x, \Delta \lambda, \Delta \nu)$

Stop if....

Advantages:

- Single loop (no inner Newton, outer central path)
- $x^{(k)}$ need not be feasible—allowed to violate $Ax = b$

$$\boxed{\begin{array}{ll} \min_{x\in\mathbb{R}^n} & f_0(x) \\ s.t. & f_i(x) \le 0 \\ & \quad i = 1..m \\ & Ax = b \\ & \quad A \in \mathbb{R}^{p\times n} \end{array}} \quad \begin{array}{l} \lambda \in \mathbb{R}^m \\ \\ \nu \in \mathbb{R}^m \end{array}$$

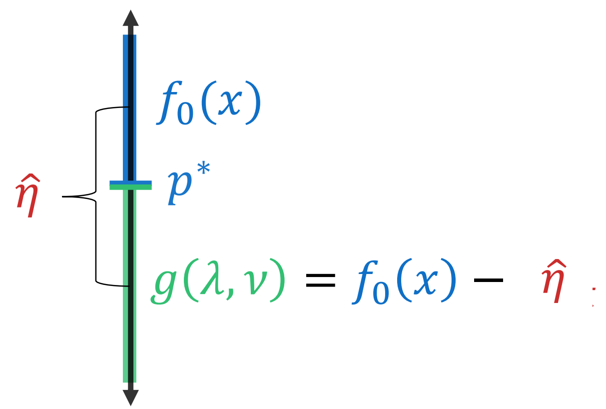
$$r_P(x) = Ax - b \in \mathbb{R}^p$$

$$r_D(x,\lambda,\nu) = \nabla_x L(x,\lambda,\nu) \in \mathbb{R}^n$$

$$r_{C(t)}(x,\lambda) = \left[\lambda_i f_i(x) + \tfrac{1}{t}\right]_{i=1..m} \in \mathbb{R}^m$$

$$r_t(x,\lambda,\nu) = \left[r_p\ r_D\ r_{C(t)}\right] \in \mathbb{R}^{n+m+p}$$

- $r_P(x) = 0$ ➔ primal feasible
- $r_D(x,\lambda,\nu) = 0$ ➔ x minimizes $L(x,\lambda,\nu)$
  ➔ $g(\lambda,\nu) = L(x,\lambda,\nu) > -\infty$ ➔ dual feasible
- $r_{C(t)}(x,\lambda) = 0$ and ALSO $r_P = r_D = 0$
  ➔ $g(\lambda,\nu) = f_0(x) + \sum \lambda_i f_i(x) + \cancel{\nu^\top(Ax-b)} = f_0(x) - {}^m/_t$
- Both $r_P = r_D = 0$ even without $r_C = 0$
  ➔ $g(\lambda,\nu) = f_0(x) + \underbrace{\sum \lambda_i f_i(x)}_{-\hat{\eta}(x,\lambda)}$



Conclusion: If $r_P = r_D = 0$, $\hat{\eta}(x,\lambda) \stackrel{\text{def}}{=} -\sum \lambda_i f_i(x)$ bounds the suboptimality

Primal-Dual Interior Point Method

Init: $x^{(0)}, \lambda^{(0)}, \nu^{(0)}$ s.t. $f_i\left(x^{(0)}\right) < 0$, $f_0\left(x^{(0)}\right) < \infty, \lambda^{(0)} > 0$

Iterate:

Calculate $\hat{\eta} = -\sum_i \lambda_i^{(k)} f_i(x^{(k)})$

$t^{(k)} = {}^{\mu m}/_{\hat{\eta}}$ (for some parameter $\mu > 1$)

Solve linearized $t^{(k)}$-relaxed KKT:

$$\begin{bmatrix} \nabla^2 f\left(x^{(k)}\right) + \sum \lambda_i^{(k)} \nabla^2 f_i\left(x^{(k)}\right) & \left(\nabla f_i\left(x^{(k)}\right)\right)_i & A^\top \\ \left(\lambda_i^{(k)} \nabla f_i\left(x^{(k)}\right)^\top\right)_i & diag\left(f_i\left(x^{(k)}\right)\right) & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta \nu \end{bmatrix} = - \begin{bmatrix} r_d\left(x^{(k)}, \lambda^{(k)}, \nu^{(k)}\right) \\ r_{c\left(t^{(k)}\right)}\left(x^{(k)}, \lambda^{(k)}\right) \\ r_p\left(x^{(k)}\right) \end{bmatrix}$$

Set stepsize s by backtracking linesearch on $\left\|r_{t^{(k)}}\right\|$,

ensuring $f_i(x) < 0$ and $\lambda_i > 0$

$\left(x^{(k+1)}, \lambda^{(k+1)}, \nu^{(k+1)}\right) \leftarrow \left(x^{(k+1)}, \lambda^{(k+1)}, \nu^{(k+1)}\right) + s(\Delta x, \Delta \lambda, \Delta \nu)$

Stop if $\|r_P\| < \epsilon_{feas}$, $\|r_D\| < \epsilon_{feas}$, and $\hat{\eta} < \epsilon$

- Single loop (no inner Newton, outer central path)
- $x^{(k)}$ need not be feasible—allowed to violate $Ax = b$
  (can use this to rewrite problem so that original $f_i(x) < 0$ violated)

Avoiding Phase I with the Primal Dual Interior Point Method

- The P/D method allows us to start at $Ax^{(0)} \neq b$, but we still need $f_i\left(x^{(0)}\right) < 0$ and $f_0\left(x^{(0)}\right) < \infty$

$$\boxed{\begin{array}{ll} \min\limits_{x \in \mathbb{R}^n} & f_0(x) \\ s.t. & f_i(x) \leq 0 \\ & Ax = b \end{array}} \Longrightarrow \boxed{\begin{array}{ll} \min\limits_{x \in \mathbb{R}^n, s \in \mathbb{R}} & f_0(x) \\ s.t. & f_i(x) \leq s \\ & Ax = b \\ & s = 0 \end{array}}$$

Initialize to any $x \in dom(f_0, f_1, \dots, f_m)$

Set $s = \max\limits_i f_i(x) + 1$

Avoiding Phase I with the Primal Dual Interior Point Method

- The P/D method allows us to start at $Ax^{(0)} \neq b$, but we still need $f_i\left(x^{(0)}\right) < 0$ and $f_0\left(x^{(0)}\right) < \infty$

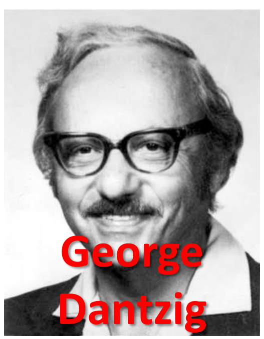
$$
\boxed{\begin{array}{ll}
\min\limits_{x \in \mathbb{R}^n} & f_0(x) \\
s.t. & f_i(x) \leq 0 \\
 & Ax = b
\end{array}}
\Longrightarrow
\boxed{\begin{array}{ll}
\min\limits_{\substack{x \in \mathbb{R}^n, s \in \mathbb{R} \\ x_i \in \mathbb{R}^n}} & f_0(x) \\
s.t. & f_i(x_i) \leq s \\
 & Ax = b \\
 & s = 0 \\
 & x_i = x \quad \forall i = 1..m
\end{array}}
$$

Initialize to any $x \in dom(f_0), x_i \in dom(f_i)$

Set $s = \max\limits_i f_i(x_i) + 1$

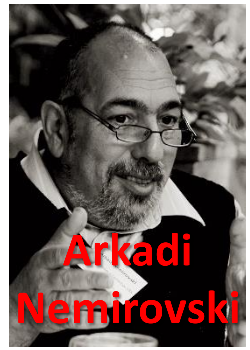
Formulation of LP

1939



Simplex

1947



Ellipsoid Method

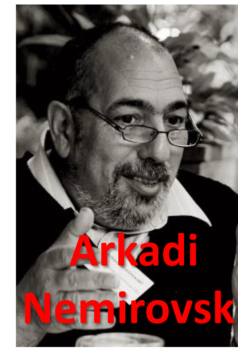
1972

Ellipsoid is Poly time for LP

1979

IP for LP

1984



General IP

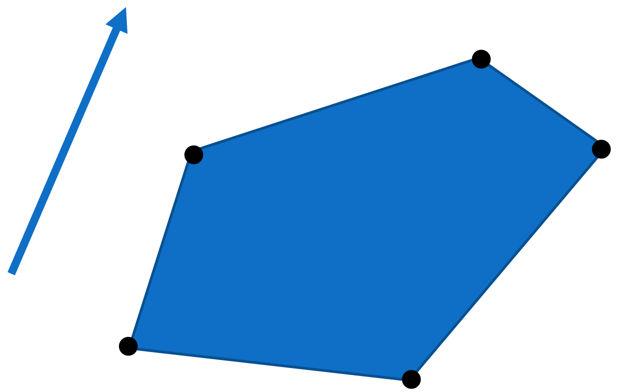
1989 -1994

The Simplex Method

$$\boxed{\begin{array}{ll} \min\limits_{x \in \mathbb{R}^n} & c^\top x \\ s.t. & Ax \leq b \end{array}}$$

$$\boxed{\begin{array}{ll} \min\limits_{x \in \mathbb{R}^n} & c^\top x \\ s.t. & Ax \leq b \end{array}}$$

- For a linear program: optimum always obtained on a vertex



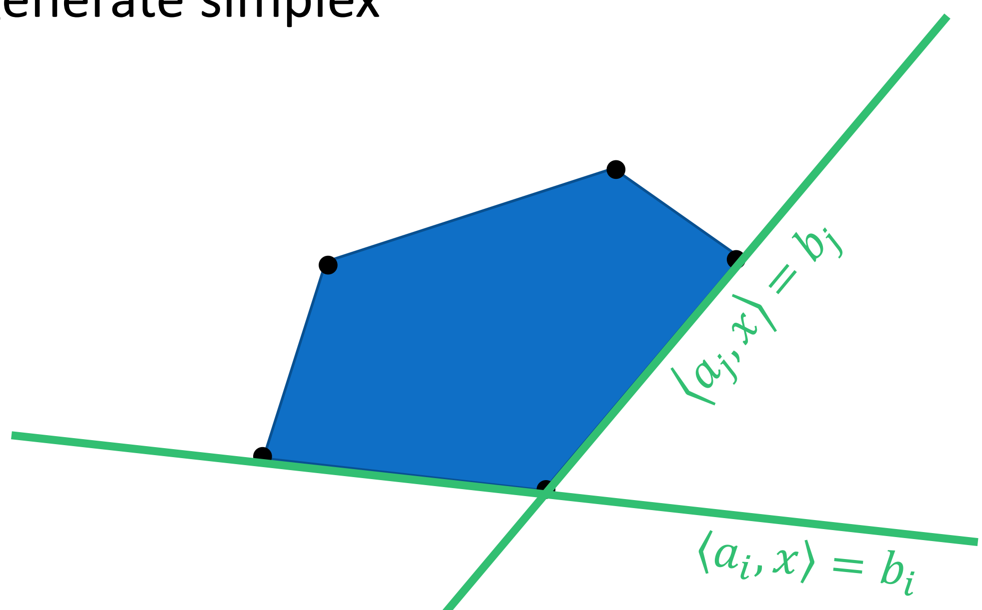
$$\boxed{\begin{array}{ll} \min_{x \in \mathbb{R}^n} & c^\top x \\ s.t. & Ax \leq b \end{array}}$$

- For a linear program: optimum always obtained on a vertex
- What's a vertex?
- For any feasible $\tilde{x}$ consider set of active constraints: $S = \{i | \langle a_i, \tilde{x} \rangle = b_i\}$
- $rank(A_s) = n$ ➔ $\tilde{x}$ is a vertex

  Vertex defined uniquely by S as solution to $A_S x = b_S$
- $A_S \in \mathbb{R}^{n \times n}$ full rank (n linearly independent constraints tight)

  ➔ we say $\tilde{x}$ is a non-degenerate simplex

  ➔ $\tilde{x} = A_S^{-1} b_S$



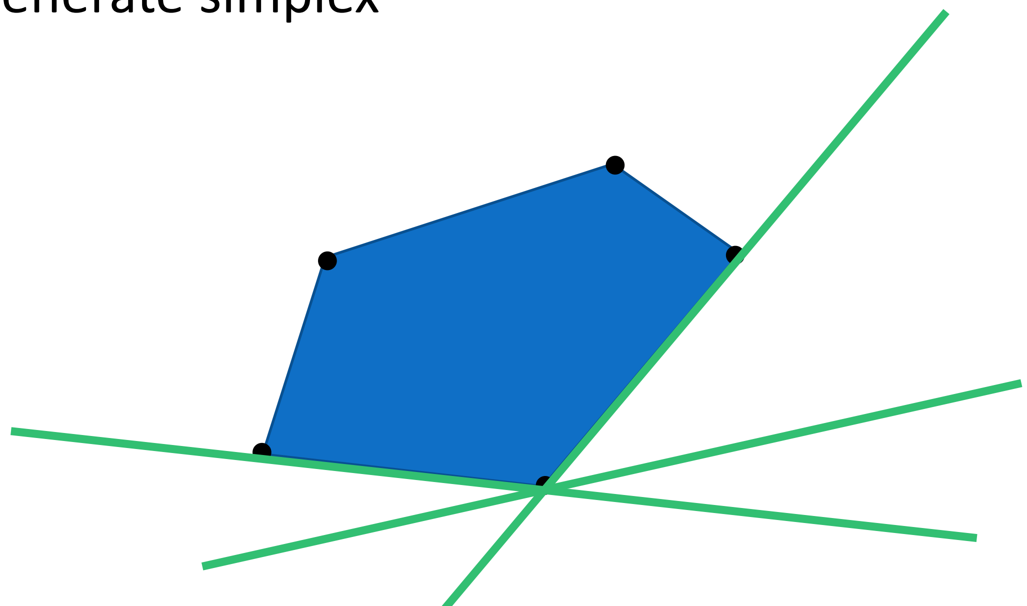
$$\boxed{\begin{array}{ll} \min_{x \in \mathbb{R}^n} & c^\top x \\ s.t. & Ax \le b \end{array}}$$

- For a linear program: optimum always obtained on a vertex
- What's a vertex?
- For any feasible $\tilde{x}$ consider set of active constraints: $S = \{i | \langle a_i, \tilde{x} \rangle = b_i\}$
- $rank(A_s) = n$ ➔ $\tilde{x}$ is a vertex

  Vertex defined uniquely by S as solution to $A_S x = b_S$
- $A_S \in \mathbb{R}^{n \times n}$ full rank (n linearly independent constraints tight)

  ➔ we say $\tilde{x}$ is a non-degenerate simplex

  ➔ $\tilde{x} = A_S^{-1} b_S$



$$\boxed{\begin{array}{ll} \min_{x\in\mathbb{R}^n} & c^\top x \\ s.t. & Ax \le b \end{array}}$$

- For a linear program: optimum always obtained on a vertex
- What's a vertex?
- For any feasible $\tilde{x}$ consider set of active constraints: $S = \{i | \langle a_i, \tilde{x}\rangle = b_i\}$
- $rank(A_s) = n$ ➔ $\tilde{x}$ is a vertex

  Vertex defined uniquely by S as solution to $A_S x = b_S$
- $A_S \in \mathbb{R}^{n\times n}$ full rank (n linearly independent constraints tight)

  ➔ we say $\tilde{x}$ is a non-degenerate simplex

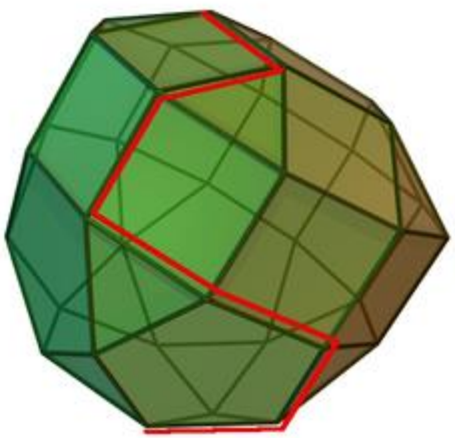
  ➔ $\tilde{x} = A_S^{-1} b_S$

We will mostly assume today A is in general position (i.e. any n rows of A are linearly independent)

➔ all vertices are non-degenerate

$$\boxed{\begin{array}{ll} \min_{x \in \mathbb{R}^n} & c^\top x \\ s.t. & Ax \leq b \end{array}}$$

- Can limit our attention to vertices
- Could be exponentially many! (e.g. $\{x | -1 \leq x_i \leq 1\}$)
- If a vertex is not optimal, there is an edge from it to a better vertex
- Simplex Method:
  - Start from some feasible vertex
  - Walk along edges of polytope, improving objective
  - End up in optimal vertex



- Maintain set S of active constraints, and current vertex $x = A_S^{-1} b_S$
- At each iteration, remove one constraint $q \in S$ and replace with another (move to a neighboring vertex by replacing one constraint in S)

$$x^+ = x + t\Delta x \quad \text{where} \quad A_S \Delta x = -e_q \;\; (\text{i.e. } \Delta x = -A_S^{-1} e_q)$$

$$(\text{i.e. } a_q^\top \Delta x = -1 \text{ while } A_{S\backslash q} \Delta x = 0\ )$$

- This ensures:

$$A_S x^+ = A_S(x + t\Delta x) = b_S - t e_q \leq b_S$$

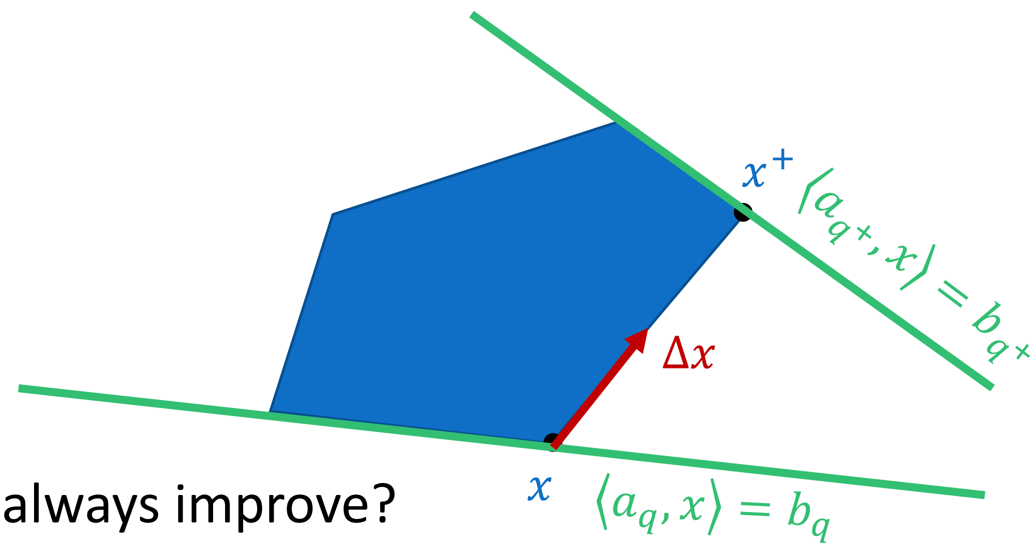
- On other vertices:

$$A_{\bar S} x^+ = \underbrace{A_{\bar S} x}_{< b_{\bar S}} + t A_{\bar S} \Delta x \leq b_{\bar S}$$

for small enough t

$$t = \min_{i \; s.t. a_i^\top \Delta x > 0} \frac{b_i - a_i^\top x}{a_i^\top \Delta x}$$

Add $q^+ = \arg\max$



Which q do we remove? Do we always improve?

$$\boxed{\begin{array}{ll}\min\limits_{x\in\mathbb{R}^n} & c^\top x \\ s.t. & Ax \le b\end{array}}$$

$$L(x,\lambda) = c^\top x + \lambda^\top (Ax - b)$$

- KKT:

$$Ax \le b \qquad \lambda \ge 0 \qquad 0 = \nabla_x L(x,\lambda) = c + A^\top \lambda \qquad \lambda_i(\langle a_i, x\rangle - b_i) = 0$$

- Construct λ:

$$\boxed{c = -A^\top \lambda = -A_S^\top \lambda_S - \cancel{A_{\bar{S}}^\top \lambda_{\bar{S}}}}$$

$$\lambda_S = -(A_S^\top)^{-1} c \qquad \lambda_{\bar{S}} = 0$$

- Change in objective after removing constraint q s.t. $\lambda_q < 0$

$$c^\top x^+ - c^\top x = c^\top (x + t\Delta x) - c^\top x = t c^\top \Delta x = t\lambda_S A_S A_S^{-1} e_q = t\lambda_q < 0$$

$$\boxed{c^\top = (-A_S^\top \lambda_S)^\top} \qquad \boxed{\Delta x = -A_S^{-1} e_q}$$

- If $\lambda_q \ge 0$ ➔ KKT satisfied, x optimal.
- Otherwise: remove $q \in S$ s.t. $\lambda_q < 0$

The Simplex Method

Init: Feasible vertex $x^{(0)}$ with active set $S^{(0)}$

Iterate:

Calculate $\lambda_S = -(A_S^\top)^{-1} c$

If $\lambda_S \geq 0$, then stop

$q = \arg\min_i \lambda_i$

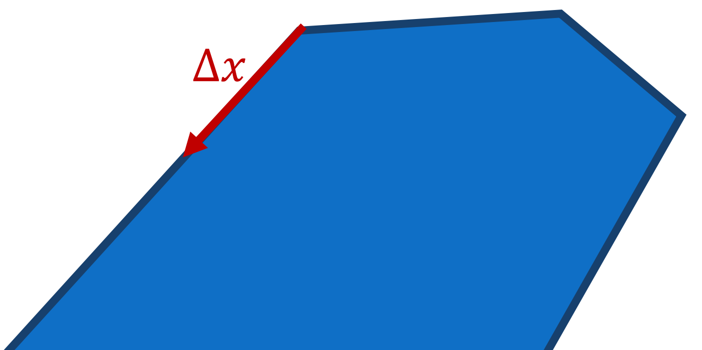
$\Delta x = -A_S^{-1} e_q$

If $A\Delta x \leq 0$, then declare unbounded ($p^* = -\infty$)

$t = \min_{i \text{ s.t.} a_i^\top \Delta x > 0} \frac{b_i - a_i^\top x}{a_i^\top \Delta x}$ and $q^+ = \arg\min$

$x^{(k+1)} \leftarrow x^{(k)} + t\Delta x$

$S^{(k+1)} \leftarrow S^{(k)} - \{q\} + \{q^+\}$



The Simplex Method

Init: Feasible vertex $x^{(0)}$ with active set $S^{(0)}$

Iterate:

Calculate $\lambda_S = -(A_S^\top)^{-1} c$

If $\lambda_S \geq 0$, then stop

$q = \arg\min_i \lambda_i$

$\Delta x = -A_S^{-1} e_q$

If $A\Delta x \leq 0$, then declare unbounded ($p^* = -\infty$)

$t = \min_{i \text{ s.t.} a_i^\top \Delta x > 0} \frac{b_i - a_i^\top x}{a_i^\top \Delta x}$ and $q^+ = \arg\min$

$x^{(k+1)} \leftarrow x^{(k)} + t\Delta x$

$S^{(k+1)} \leftarrow S^{(k)} - \{q\} + \{q^+\}$

- Runtime per iteration: $O(n^3)$

  Can be reduced to $O(n^2)$ by updating $(A_S^\top)^{-1}$ directly
- Number of iterations?

Simplex Runtime

- #Itter typically “small”
- Klee and Minty 1972: Could be 2^n, even with $O(n)$ constraints
- Spielman and Teng 2001 “smoothed analysis”: For any LP problem, if we add small random perturbations to the problem, with high probability over perturbation, $O(m)$ steps

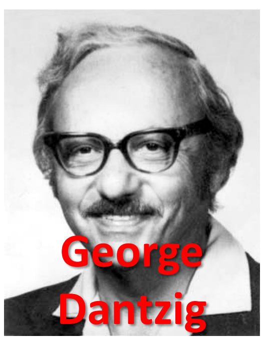
Simplex—Additional Issues

- Efficient linear algebra of pivoting
- Handling degenerate vertices
- Finding initial feasible vertex
  - Phase I method to find feasible point
  - Can then add non-tight constraints until it's a vertex
- Other "pivoting rules" (for choosing q)

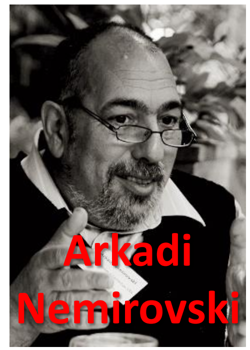
Simplex vs IP Methods

- Worst case performance can be bad
- Specific to LP, not a black-box method
- Violate $\lambda \geq 0$ (qv complimentary slackness)
  - x is always exactly feasible (not strictly feasible)
  - Work on “active set” of constraints
  - Example of “active set” method

Formulation of LP



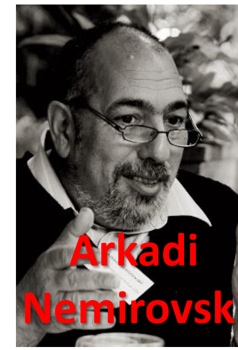
Simplex



Ellipsoid
Method

Ellipsoid is
Poly time
for LP

IP for LP



General IP

1939 1947 1972 1979 1984 1989
-1994