Convex Optimization

Stephen Boyd

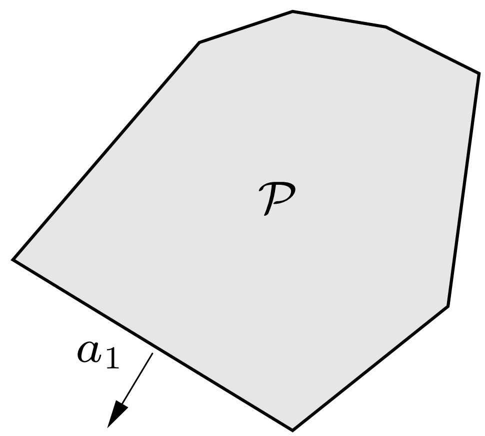
Electrical Engineering Department
Stanford University

(joint work with **Lieven Vandenberghe**, UCLA)

IAM-PIMS, Vancouver 3/15/04

Two problems

polyhedron $\mathcal{P}$ described by linear inequalities, $a_i^T x \leq b_i$, $i = 1, \ldots, m$



Problem 1: find minimum volume ellipsoid $\supseteq \mathcal{P}$

Problem 2: find maximum volume ellipsoid $\subseteq \mathcal{P}$

are these (computationally) difficult? or easy?

problem 1 is **very difficult**

- in practice
- in theory (NP-hard)

problem 2 is **very easy**

- in practice (readily solved on small computer)
- in theory (polynomial complexity)

Moral

very difficult and **very easy** problems can look **quite similar**

. . . unless you're trained to recognize the difference

Linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \le b_i, \quad i = 1, \ldots, m \end{array}$$

$c,\ a_i \in \mathbf{R}^n$ are parameters; $x \in \mathbf{R}^n$ is variable

- **easy** to solve, in theory and practice
- can solve dense problems with $n = 1000$ vbles, $m = 10000$ constraints easily; far larger for sparse or structured problems

Polynomial minimization

$$\text{minimize} \quad p(x)$$

p is polynomial of degree d; $x \in \mathbf{R}^n$ is variable

- except for special cases (e.g., $d = 2$) this is a **very difficult problem**
- even sparse problems with size $n = 20$, $d = 10$ are essentially intractable
- all algorithms known to solve this problem require effort exponential in n

Moral

- a problem can **appear*** **hard**, but **be easy**
- a problem can **appear*** **easy**, but **be hard**

* to the untrained eye

What makes a problem easy or hard?

classical view:

- **linear** is easy
- **nonlinear** is hard(er)

What makes a problem easy or hard?

emerging (and correct) view:

. . . the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.

— R. Rockafellar, SIAM Review 1993

Convex optimization

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_1(x) \leq 0, \ldots, f_m(x) \leq 0, \quad Ax = b \end{array}$$

$x \in \mathbf{R}^n$ is optimization variable; $f_i : \mathbf{R}^n \to \mathbf{R}$ are **convex**:

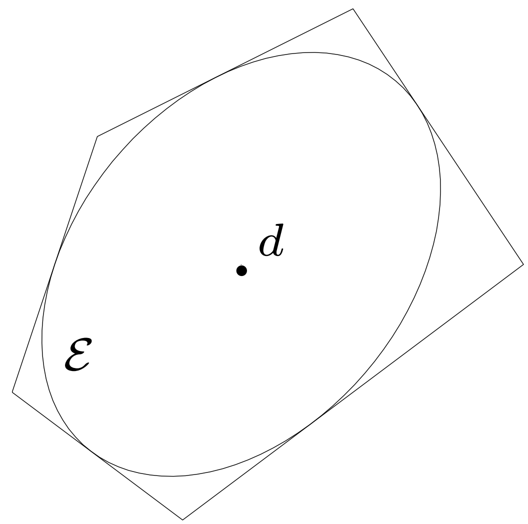
$$f_i(\lambda x + (1 - \lambda)y) \leq \lambda f_i(x) + (1 - \lambda) f_i(y)$$

for all x, y, $0 \leq \lambda \leq 1$

- includes least-squares, linear programming, maximum volume ellipsoid in polyhedron, and **many others**
- convex problems are **fundamentally tractable**

Maximum volume ellipsoid in polyhedron

- polyhedron: $\mathcal{P} = \{x \mid a_i^T x \leq b_i,\ i = 1, \ldots, m\}$
- ellipsoid: $\mathcal{E} = \{By + d \mid \|y\| \leq 1\}$, with $B = B^T \succ 0$



maximum volume $\mathcal{E} \subseteq \mathcal{P}$, as convex problem in variables B, d:

$$\begin{array}{ll} \text{maximize} & \log\det B \\ \text{subject to} & B = B^T \succ 0, \quad \|Ba_i\| + a_i^T d \leq b_i, \quad i = 1, \ldots, m \end{array}$$

Moral

- it's not easy to recognize convex functions and convex optimization problems
- **huge benefit**, though, when you do

Convex Analysis and Optimization

Convex analysis & optimization

nice properties of convex optimization problems known since 1960s

- local solutions are global
- duality theory, optimality conditions
- simple solution methods like alternating projections

convex analysis well developed by 1970s *Rockafellar*

- separating & supporting hyperplanes
- subgradient calculus

What's new (since 1990 or so)

- primal-dual interior-point (IP) methods
  extremely efficient, handle nonlinear large scale problems, polynomial-time complexity results, software implementations

- new standard problem classes
  generalizations of LP, with theory, algorithms, software

- extension to generalized inequalities
  semidefinite, cone programming

Applications and uses

- lots of engineering applications
  control, combinatorial optimization, signal processing, circuit design, communications, . . .

- robust optimization
  robust versions of LP, least-squares, other problems

- relaxations and randomization
  provide bounds, heuristics for solving hard ($e.g.$, combinatorial optimization) problems

Recent history

- 1984–97: interior-point methods for LP
  - 1984: Karmarkar's interior-point LP method
  - theory *Ye, Renegar, Kojima, Todd, Monteiro, Roos, . . .*
  - practice *Wright, Mehrotra, Vanderbei, Shanno, Lustig, . . .*
- 1988: Nesterov & Nemirovsky's self-concordance analysis
- 1989–: LMIs and semidefinite programming in control
- 1990–: semidefinite programming in combinatorial optimization
  Alizadeh, Goemans, Williamson, Lovasz & Schrijver, Parrilo, . . .
- 1994: interior-point methods for nonlinear convex problems
  Nesterov & Nemirovsky, Overton, Todd, Ye, Sturm, . . .
- 1997–: robust optimization *Ben Tal, Nemirovsky, El Ghaoui, . . .*

New Standard Convex Problem Classes

Some new standard convex problem classes

- second-order cone program (SOCP)
- geometric program (GP) (and entropy problems)
- semidefinite program (SDP)

for these new problem classes we have

- complete duality theory, similar to LP
- good algorithms, and robust, reliable software
- wide variety of new applications

Second-order cone program

second-order cone program (SOCP) has form

$$
\begin{array}{ll}
\text{minimize} & c_0^T x \\
\text{subject to} & \|A_i x + b_i\|_2 \leq c_i^T x + d_i, \quad i = 1, \ldots, m
\end{array}
$$

with variable $x \in \mathbf{R}^n$

- includes LP and QP as special cases
- nondifferentiable when $A_i x + b_i = 0$
- new IP methods can solve (almost) as fast as LPs

Example: robust linear program

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{Prob}(a_i^T x \leq b_i) \geq \eta, \quad i = 1, \ldots, m \end{array}$$

where $a_i \sim \mathcal{N}(\bar{a}_i, \Sigma_i)$

equivalent to

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \bar{a}_i^T x + \Phi^{-1}(\eta) \|\Sigma_i^{1/2} x\|_2 \leq 1, \quad i = 1, \ldots, m \end{array}$$

where Φ is (unit) normal CDF

robust LP is an SOCP for $\eta \geq 0.5$ ($\Phi(\eta) \geq 0$)

Geometric program (GP)

log-sum-exp function:

$$\mathbf{lse}(x) = \log\left(e^{x_1} + \cdots + e^{x_n}\right)$$

. . . a smooth **convex** approximation of the max function

geometric program:

$$\begin{array}{ll} \text{minimize} & \mathbf{lse}(A_0 x + b_0) \\ \text{subject to} & \mathbf{lse}(A_i x + b_i) \le 0, \quad i = 1, \ldots, m \end{array}$$

$A_i \in \mathbf{R}^{m_i \times n}$, $b_i \in \mathbf{R}^{m_i}$; variable $x \in \mathbf{R}^n$

Entropy problems

unnormalized negative entropy is convex function

$$-\mathbf{entr}(x) = \sum_{i=1}^{n} x_i \log(x_i/\mathbf{1}^T x)$$

defined for $x_i \geq 0$, $\mathbf{1}^T x > 0$

entropy problem:

$$\begin{array}{ll} \text{minimize} & -\mathbf{entr}(A_0 x + b_0) \\ \text{subject to} & -\mathbf{entr}(A_i x + b_i) \leq 0, \quad i = 1, \ldots, m \end{array}$$

$A_i \in \mathbf{R}^{m_i \times n}$, $b_i \in \mathbf{R}^{m_i}$

Solving GPs (and entropy problems)

- GP and entropy problems are **duals** (if we solve one, we solve the other)
- new IP methods can solve large scale GPs (and entropy problems) almost as fast as LPs
- applications in many areas:
  - information theory, statistics
  - communications, wireless power control
  - digital and analog circuit design

Semidefinite program

semidefinite program (SDP):

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x_1 A_1 + \cdots + x_n A_n \preceq B \end{array}$$

- B, A_i are symmetric matrices; variable is $x \in \mathbf{R}^n$
- $\preceq$ is matrix inequality; constraint is **linear matrix inequality** (LMI)
- SDP can be expressed as convex problem as

$$\lambda_{\max}(x_1 A_1 + \cdots + x_n A_n - B) \leq 0$$

or handled directly as **cone problem**

Early SDP applications

(around 1990 on)

- control *(many)*
- combinatorial optimization & graph theory *(many)*

More recent SDP applications

- structural optimization: *Ben-Tal, Nemirovsky, Kocvara, Bendsoe, . . .*
- signal processing: *Vandenberghe, Stoica, Lorenz, Davidson, Shaked, Nguyen, Luo, Sturm, Balakrishnan, Saadat, Fu, de Souza, . . .*
- circuit design: *El Gamal, Vandenberghe, Boyd, Yun, . . .*
- algebraic geometry:
  Parrilo, Sturmfels, Lasserre, de Klerk, Pressman, Pasechnik, . . .
- communications and information theory:
  Rasmussen, Rains, Abdi, Moulines, . . .
- quantum computing:
  Kitaev, Waltrous, Doherty, Parrilo, Spedalieri, Rains, . . .
- finance: *Iyengar, Goldfarb, . . .*

Moment problems

$$\mu_i = \mathbf{E}\, t^i, \quad i = 1, \ldots, 2n$$

for some probability distribution on **R** if and only if

$$H(\mu) = \begin{bmatrix} 1 & \mu_1 & \cdots & \mu_{n-1} & \mu_n \\ \mu_1 & \mu_2 & \cdots & \mu_n & \mu_{n+1} \\ \vdots & \vdots & & \vdots & \vdots \\ \mu_{n-1} & \mu_n & \cdots & \mu_{2n-2} & \mu_{2n-1} \\ \mu_n & \mu_{n+1} & \cdots & \mu_{2n-1} & \mu_{2n} \end{bmatrix} \succeq 0$$

- an LMI in μ
- similar results for bounded and half-bounded intervals; trigonometric moments

Moment bounds via SDP

problem: given bounds on moments, $\underline{\mu}_i \leq \mathbf{E}\, t^i \leq \overline{\mu}_i$, find bounds on

$$\mathbf{E}\left(c_0 + c_1 t + \cdots + c_{2n} t^{2n}\right) = c^T \mu$$

$$\begin{array}{ll} \text{maximize (minimize)} & \mathbf{E}(c_0 + c_1 t + \cdots + c_{2n} t^{2n}) \\ \text{subject to} & \underline{\mu}_i \leq \mathbf{E}\, t^i \leq \overline{\mu}_i, \quad i = 1, \ldots, 2n \end{array}$$

over all probability distributions on **R**
can be expressed as SDP

$$\begin{array}{ll} \text{maximize (minimize)} & c^T \mu \\ \text{subject to} & \underline{\mu}_i \leq \mu_i \leq \overline{\mu}_i, \quad H(\mu) \succeq 0 \end{array}$$

Portfolio risk

- portfolio of n assets invested for single period
- w_i is amount of investment in asset i
- returns of assets is random vector r with mean $\overline{r}$, covariance Σ
- portfolio return is random variable $r^T w$
- mean portfolio return is $\overline{r}^T w$; variance is $V = w^T \Sigma w$

value at risk & probability of loss are related to portfolio variance V

Risk bound with uncertain covariance

now suppose:

- w is known (and fixed)
- have only partial information about Σ, *i.e.*,

$$L_{ij} \leq \Sigma_{ij} \leq U_{ij}, \quad i, j = 1, \ldots, n$$

problem: how large can portfolio variance $V = w^T \Sigma w$ be?

Risk bound via SDP

can get (tight) bound on V via SDP:

$$\begin{array}{ll} \text{maximize} & w^T \Sigma w \\ \text{subject to} & \Sigma \succeq 0 \\ & L_{ij} \leq \Sigma_{ij} \leq U_{ij} \end{array}$$

(note extra constraint $\Sigma \succeq 0$)

many extensions possible, $e.g.$, optimize portfolio w with worst-case variance limit

Risk bounding example

variance bounding with sign constraints on Σ:

$$w = \begin{bmatrix} 1 \\ 2 \\ -.5 \\ .5 \end{bmatrix}, \qquad \Sigma = \begin{bmatrix} 1 & + & + & ? \\ + & 1 & - & - \\ + & - & 1 & + \\ ? & - & + & 1 \end{bmatrix}$$

($i.e.$, $\Sigma_{12} \geq 0$, $\Sigma_{23} \leq 0$, $\ldots$)

result: maximum value of V is 10.1, with

$$\Sigma = \begin{bmatrix} 1.00 & 0.80 & 0.00 & 0.50 \\ 0.80 & 1.00 & -.58 & 0.00 \\ 0.00 & -.58 & 1.00 & 0.46 \\ 0.50 & 0.00 & 0.46 & 1.00 \end{bmatrix}$$

(which has rank 3, so LMI $\Sigma \succeq 0$ is active)

Relaxations & Randomization

Relaxations & randomization

convex optimization is increasingly used

- to find good bounds for hard (i.e., nonconvex) problems, via **relaxation**
- as a heuristic for finding good suboptimal points, often via **randomization**

Example: Boolean least-squares

Boolean least-squares problem:

$$\begin{array}{ll} \text{minimize} & \|Ax - b\|^2 \\ \text{subject to} & x_i^2 = 1, \quad i = 1, \ldots, n \end{array}$$

- basic problem in digital communications
- could check all 2^n possible values of x . . .
- an NP-hard problem, and very hard in practice
- many heuristics for approximate solution

Boolean least-squares as matrix problem

$$\begin{aligned} \|Ax - b\|^2 &= x^T A^T A x - 2b^T A x + b^T b \\ &= \mathbf{Tr}\, A^T A X - 2b^T A^T x + b^T b \end{aligned}$$

where $X = xx^T$

hence can express BLS as

$$\begin{array}{ll} \text{minimize} & \mathbf{Tr}\, A^T A X - 2b^T A x + b^T b \\ \text{subject to} & X_{ii} = 1, \quad X \succeq xx^T, \quad \text{rank}(X) = 1 \end{array}$$

. . . still a very hard problem

SDP relaxation for BLS

ignore rank one constraint, and use

$$X \succeq xx^T \iff \begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0$$

to obtain **SDP relaxation** (with variables X, x)

$$\begin{array}{ll} \text{minimize} & \mathbf{Tr}\, A^TAX - 2b^TA^Tx + b^Tb \\ \text{subject to} & X_{ii} = 1, \quad \begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0 \end{array}$$

- optimal value of SDP gives **lower bound** for BLS
- if optimal matrix is rank one, we're done

Interpretation via randomization

- can think of variables X, x in SDP relaxation as defining a normal distribution $z \sim \mathcal{N}(x, X - xx^T)$, with $\mathbf{E}\, z_i^2 = 1$
- SDP objective is $\mathbf{E}\, \|Az - b\|^2$

suggests randomized method for BLS:

- find $X^\star$, $x^\star$, optimal for SDP relaxation
- generate z from $\mathcal{N}(x^\star, X^\star - x^\star x^{\star T})$
- take $x = \mathbf{sgn}(z)$ as approximate solution of BLS
  (can repeat many times and take best one)

Example

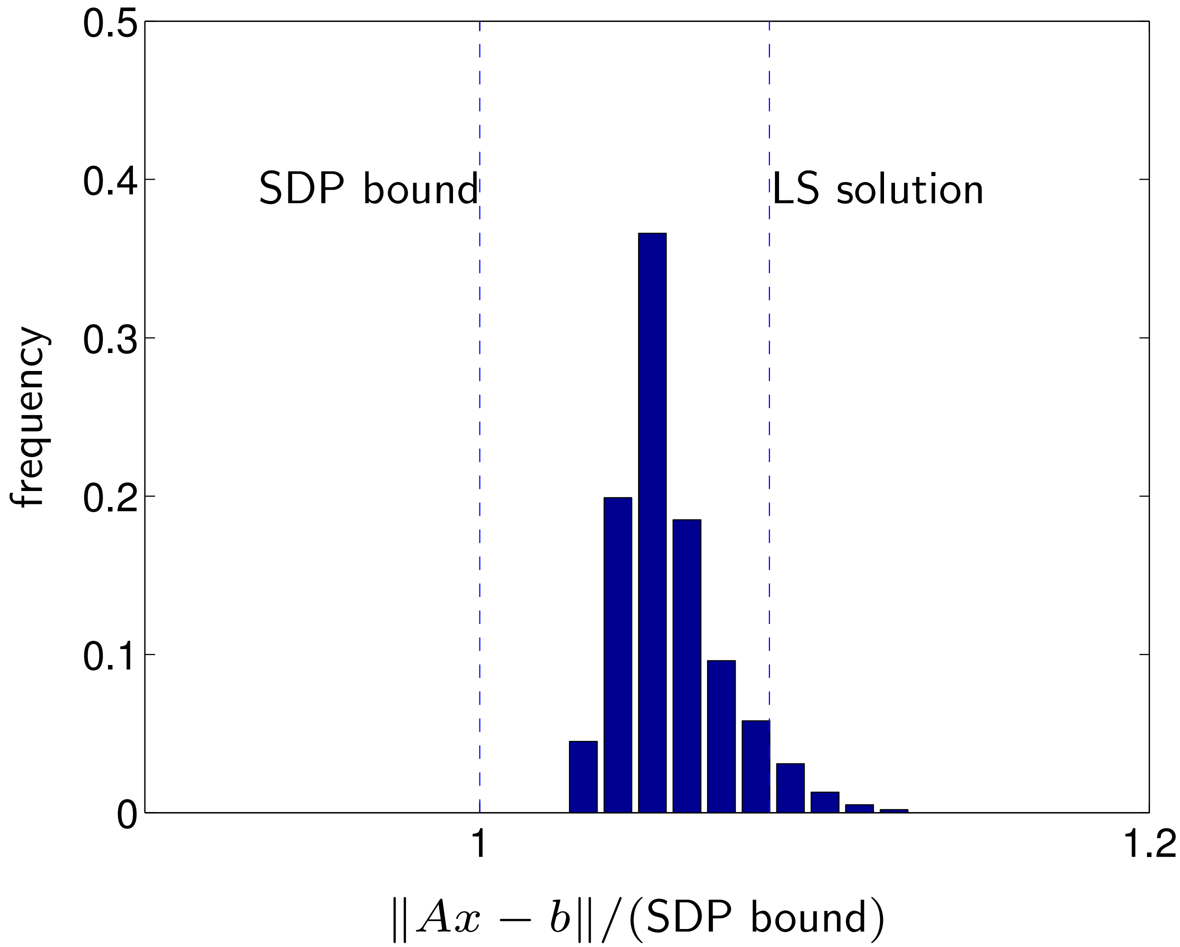
- (randomly chosen) parameters $A \in \mathbf{R}^{150\times 100}$, $b \in \mathbf{R}^{150}$
- $x \in \mathbf{R}^{100}$, so feasible set has $2^{100} \approx 10^{30}$ points

LS approximate solution: minimize $\|Ax - b\|$ s.t. $\|x\|^2 = n$, then round

yields objective 8.7% over SDP relaxation bound

randomized method: (using SDP optimal distribution)

- best of 20 samples: 3.1% over SDP bound
- best of 1000 samples: 2.6% over SDP bound


0.5
0.4
0.3
0.2
0.1
0
frequency
SDP bound
LS solution
1
1.2
$\|Ax - b\|/(\text{SDP bound})$

Interior-Point Methods

Interior-point methods

- handle linear and **nonlinear** convex problems *Nesterov & Nemirovsky*
- based on Newton's method applied to 'barrier' functions that trap x in **interior** of feasible region (hence the name IP)
- worst-case complexity theory: # Newton steps $\sim \sqrt{\text{problem size}}$
- in practice: # Newton steps between 20 & 50 (!)
  — over wide range of problem dimensions, type, and data
- 1000 variables, 10000 constraints feasible on PC; far larger if structure is exploited
- readily available (commercial and noncommercial) packages

Log barrier

for convex problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \end{array}$$

we define **logarithmic barrier** as

$$\phi(x) = -\sum_{i=1}^{m} \log(-f_i(x))$$

- ϕ is convex, smooth on interior of feasible set
- $\phi \to \infty$ as x approaches boundary of feasible set

Central path

central path is curve

$$x^\star(t) = \underset{x}{\operatorname{argmin}} \left(t f_0(x) + \phi(x) \right), \qquad t \geq 0$$

- $x^\star(t)$ is strictly feasible, *i.e.*, $f_i(x) < 0$
- $x^\star(t)$ can be computed by, *e.g.*, Newton's method
- intuition suggests $x^\star(t)$ converges to optimal as $t \to \infty$
- using duality can prove $x^\star(t)$ is m/t-suboptimal

Barrier method

a.k.a. **path-following method**

given strictly feasible x, $t > 0$, $\mu > 1$

repeat

1. compute $x := x^\star(t)$

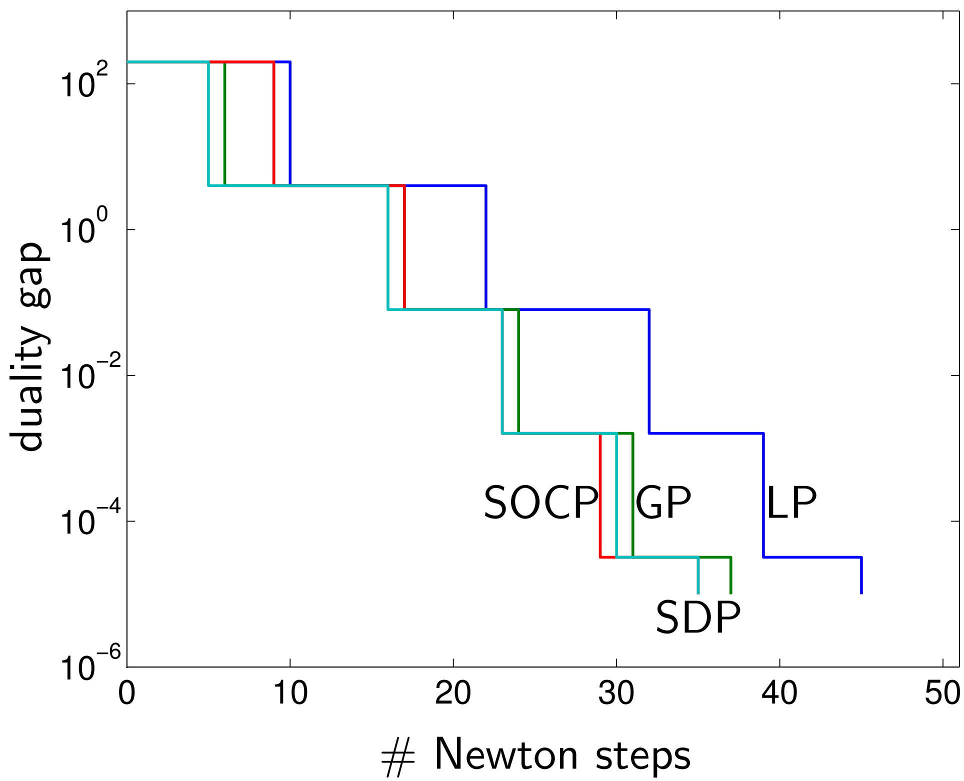
   (using Newton's method, starting from x)

2. **exit if** $m/t <$ tol

3. $t := \mu t$

duality gap reduced by μ each outer iteration

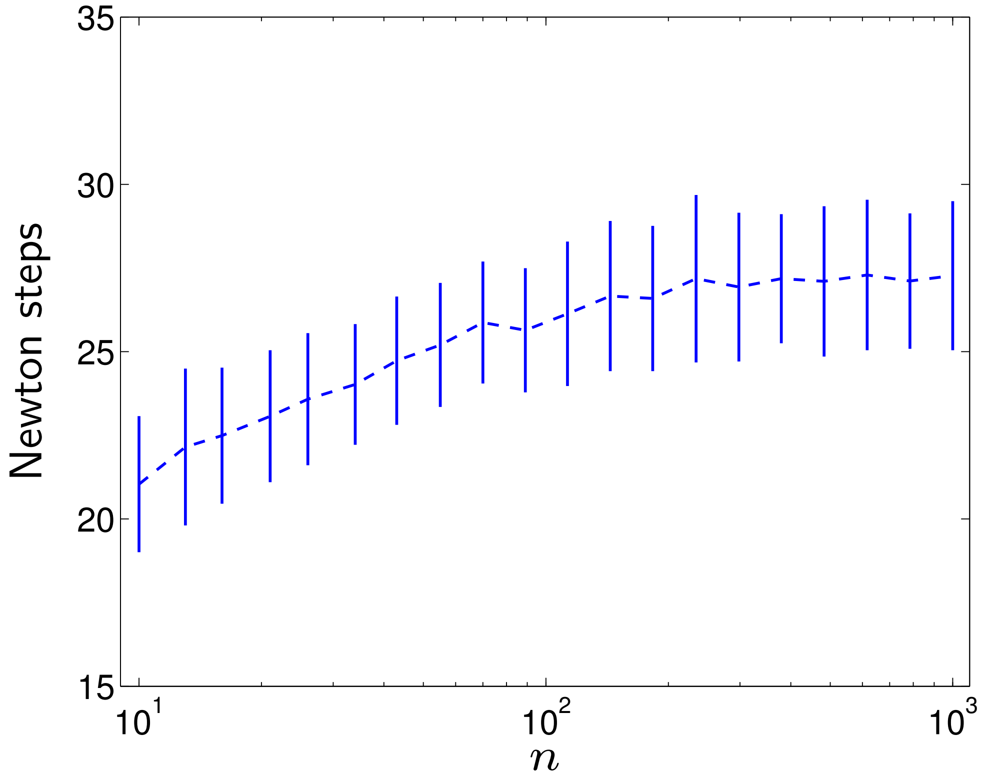
Typical convergence of IP method



LP, GP, SOCP, SDP with 100 variables

Typical effort versus problem dimensions

- LPs with n vbles, $2n$ constraints
- 100 instances for each of 20 problem sizes
- avg & std dev shown



Computational effort per Newton step

- Newton step effort dominated by solving linear equations to find primal-dual search direction
- equations inherit structure from underlying problem
- equations same as for least-squares problem of similar size and structure

conclusion:

we can solve a **convex problem** with about the same effort as solving **20–50 least-squares problems**

Problem structure

common types of structure:

- sparsity
- state structure
- Toeplitz, circulant, Hankel; displacement rank
- Kronecker, Lyapunov structure
- symmetry

Exploiting sparsity

- well developed, since late 1970s
- direct (sparse factorizations) and iterative methods (CG, LSQR)
- standard in general purpose LP, QP, GP, SOCP implementations
- can solve problems with 10^5, 10^6 vbles, constraints
  (depending on sparsity pattern)

Other interior-point methods

more sophisticated IP algorithms

- primal-dual, incomplete centering, infeasible start
- use same ideas, $e.g.$, central path, log barrier
- readily available (commercial and noncommercial packages)

typical performance: $20 - 50$ Newton steps (!)
— over wide range of problem dimensions, problem type, and problem data

Conclusions

Conclusions

convex optimization

- theory fairly mature; practice has advanced tremendously last decade
- qualitatively different from general nonlinear programming
- cost only $30\times$ more than least-squares, but far more expressive
- **lots of applications** still to be discovered

Some references

- Semidefinite Programming, *SIAM Review 1996*
- Applications of Second-order Cone Programming, *LAA 1999*
- Linear Matrix Inequalities in System and Control Theory, *SIAM 1994*
- Interior-point Polynomial Algorithms in Convex Programming, *SIAM 1994, Nesterov & Nemirovsky*
- Lectures on Modern Convex Optimization, *SIAM 2001, Ben Tal & Nemirovsky*

Shameless promotion

Convex Optimization, *Boyd & Vandenberghe*

- published by Cambridge University Press 2003; ready soon
- complete text available now (and in future) at
  `www.stanford.edu/~boyd/cvxbook.html`