MOPTA 2005

How good (how bad) are Simplex and Interior Point Methods? What next?

Tamás Terlaky

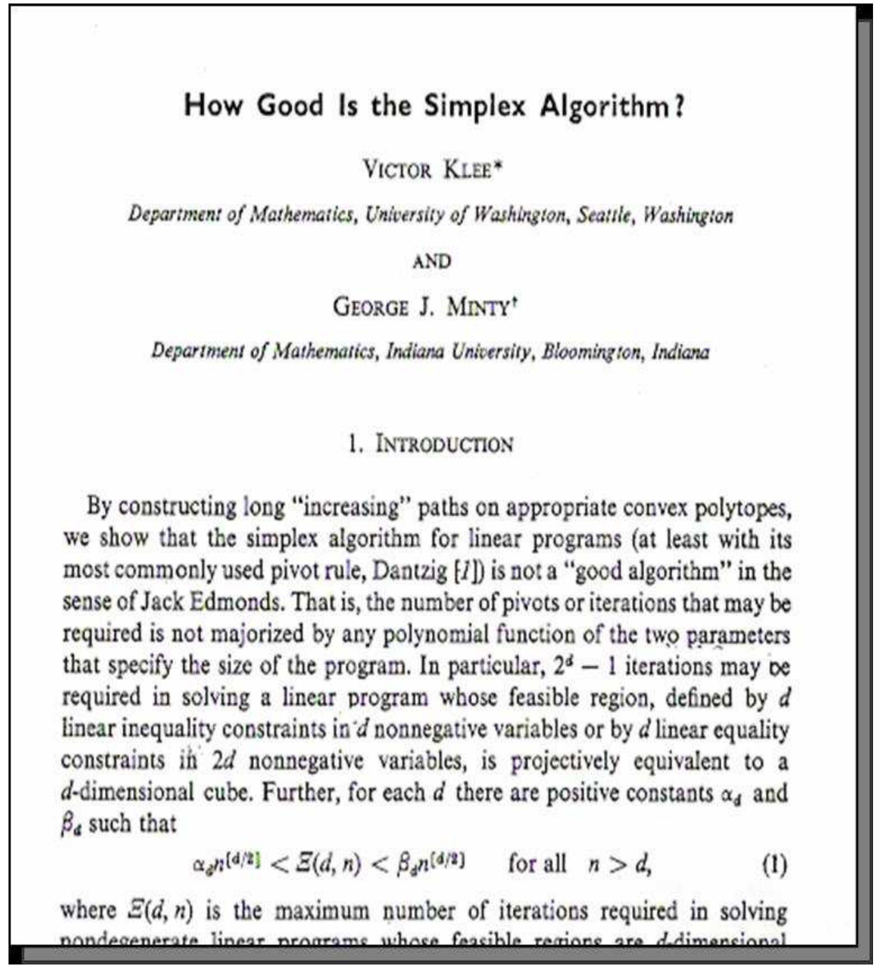
AdvOL

AdvOL, Department of Computing and Software
McMaster University, Hamilton, Ontario

July, 2005, Windsor

Outline

- Introduction
- Simplex and
  Interior Point Methods
  •State of the art–Theory
  •State of the art–software
- The Klee–Minty example
- What is next? – Today
- What is next?
  New ideas wanted!
- Conclusions



How Good Is the Simplex Algorithm?

VICTOR KLEE*

Department of Mathematics, University of Washington, Seattle, Washington

AND

GEORGE J. MINTY†

Department of Mathematics, Indiana University, Bloomington, Indiana

1. INTRODUCTION

By constructing long "increasing" paths on appropriate convex polytopes, we show that the simplex algorithm for linear programs (at least with its most commonly used pivot rule, Dantzig [1]) is not a "good algorithm" in the sense of Jack Edmonds. That is, the number of pivots or iterations that may be required is not majorized by any polynomial function of the two parameters that specify the size of the program. In particular, $2^d - 1$ iterations may be required in solving a linear program whose feasible region, defined by d linear inequality constraints in d nonnegative variables or by d linear equality constraints in $2d$ nonnegative variables, is projectively equivalent to a d-dimensional cube. Further, for each d there are positive constants α_d and β_d such that

$$\alpha_d n^{[d/2]} < \Xi(d, n) < \beta_d n^{[d/2]} \quad \text{for all} \quad n > d, \qquad (1)$$

where $\Xi(d, n)$ is the maximum number of iterations required in solving

Some Milestones in the History of LO

- 1947 simplex – Dantzig – still very efficient
- **1972 exponential example – Klee–Minty – theoretical**
- 1979 ellipsoid – Khachian – not efficient in practice
- 1984 projective IPM – Karmarkar – efficient in practice!?
- 1989 $O(n^3L)$ for IPMs – Renegar, Gonzaga, Roos, Vial
  —- best complexity so far
- 1989 Primal–Dual IPMs – Kojima ... – dominant since then
- 1989 Self-Concordant barrier – Nesterov–Nemirovskii
  —- extensions to smooth convex opt.
- 1992 SDO and SOCO – Alizadeh, Nesterov–Nemirovskii
  —- new applications, approximations, software
- 1996-2000 Volumetric center method – Vaidya, Anstreicher
- **2004 Klee-Minty example for IPMs**

Standard form LO: Fundamentals

$$(P) \quad \min \left\{ c^T x \ : \ Ax = b, \ \ x \geq 0 \right\},$$
$$(D) \quad \max \left\{ b^T y \ : \ A^T y + s = c, \ \ s \geq 0 \right\}.$$

A is an $m \times n$ matrix with rank$(A) = m$.

Proposition 1 (Weak duality). *If $x \in \mathbb{R}^n$ is primal, $y \in \mathbb{R}^m$ is dual feasible then* $c^T x \geq b^T y$, *where the equality is satisfied iff $x^T s = 0$.*

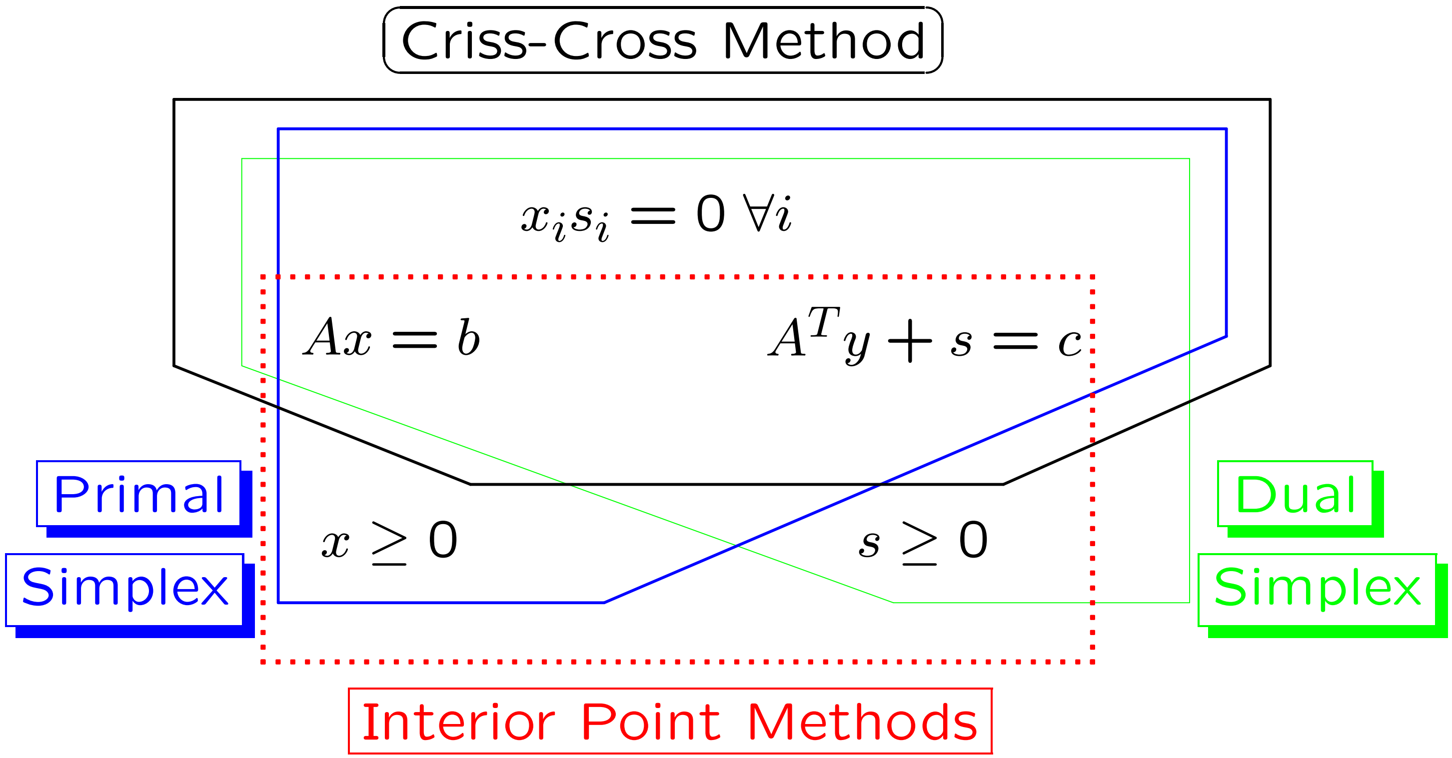
Corollary 1 (Complementarity). *If $x \in \mathbb{R}^n$ is primal, $y \in \mathbb{R}^m$ is dual feasible and $x^T s = 0$ then x and y are primal and dual optimal, respectively.*

Optimality: Optimal solutions can be given as the set of solutions of the system:

$$Ax = b, \quad x \geq 0 \qquad\qquad A^T y + s = c, \quad s \geq 0$$
$$x_i s_i = 0 \ \forall\, i$$

Linear Optimization: Fundamentals

Standard form LO model: Optimality Conditions



Algorithmic concepts

Notes on the Simplex Method

Consider the standard form LO problem

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax = b \\ & x \geq 0, \end{aligned}$$

where A has full row rank.

Simplex methods

- start from a feasible basis,
- by using a pivot rule,
- find an optimal solution (after finite number of iterations).
- Most variants are known to be exponential, nevertheless very efficient implementations exist.
- Advanced use in Mixed integer opt.; Sensitivity Analysis.

Short admissible pivot sequence exist ($\leq n$ pivot steps)

The Primal–Dual LO Problems, Central Path

The primal-dual LO problems is given as:

$$\begin{aligned} \min \quad & c^T x \\ Ax &= b, \quad x \geq 0, \end{aligned} \qquad\qquad \begin{aligned} \max \quad & b^T y \\ A^T y + s &= c, \quad s \geq 0, \end{aligned}$$

where $c, x, s \in \mathcal{R}^n,\ b, y \in \mathcal{R}^m,\ A \in \mathcal{R}^{m \times n}$, $\text{rank}(A) = m$.

Optimality conditions and the central path are given as:

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= 0, \end{aligned} \qquad\qquad \begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= \mu e, \end{aligned}$$

where $e = (1, \ldots, 1)^T \in \mathcal{R}^n$.

We assume that the Interior Point Condition holds.

Primal-Dual Search-directions for LO

The central path and the Classical Newton direction:

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= \mu e. \end{aligned} \qquad \begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu e - xs, \end{aligned}$$

Scaled Newton direction:

$$\begin{aligned} \bar{A} p_x &= 0, \\ \bar{A}^T \Delta y + p_s &= 0, \\ p_x + p_s &= v^{-1} - v \end{aligned}$$

Proximity Functions:

$$\begin{aligned} \Psi(v) &= \sum_{i=1}^{n} \left(\frac{v_i^2 - 1}{2} - \log v_i \right) \\ \Psi(v) &= \frac{1}{2} \| v - v^{-1} \|^2. \end{aligned}$$

where $\bar{A} = \frac{1}{\mu} A V^{-1} X$, $V = \operatorname{diag}(v)$, $X = \operatorname{diag}(x)$ with

$$v := \sqrt{\frac{xs}{\mu}}, \quad v^{-1} := \sqrt{\frac{\mu}{xs}}, \quad p_x := \frac{v\Delta x}{x}, \quad p_s := \frac{v\Delta s}{s}.$$

Polynomial Complexity of Small and Large update IPMs

Method	Large update	Small update	
θ	$1 - 1/100$	$1/\sqrt{n}$	
Iter. bound	$\mathcal{O}(n \log \frac{n}{\epsilon})$	$\mathcal{O}(\sqrt{n} \log \frac{n}{\epsilon})$	
Performance	Efficient	Very poor	
SR-Method	SR-Large	SR-Small	SR-Large $q = \log n$
θ	$1 - 1/100$	$1/\sqrt{n}$	constant
Iter. bound	$\mathcal{O}(q\, n^{\frac{q+1}{2q}} \log \frac{n}{\epsilon})$	$\mathcal{O}(\sqrt{n} \log \frac{n}{\epsilon})$	$\mathcal{O}(\sqrt{n} \log n \log \frac{n}{\epsilon})$
Performance	Efficient	Very poor	Efficient

"Almost" constant (< 100) number of iterations in practice!

Bixby: 10^6 speedup in a decade in solving LO problems

What made this major advance possible?

Advances in Computers and Software

Computers

- **processor speed**
- **memory**
- disk space
- floating point arithmetic
- architecture (cash ...)

Software component/Algorithms

- presolve
- dual simplex and steepest edge
- pricing/shifting/Harris test
- IPMs, predictor-corrector
- **SPARSE LINEAR ALGEBRA**
- sparse factorizations

CPLEX, XPRESS-MP, FORTMP, LOQO, MOSEK, LIPSOL, Pcx, McIPM

Best Speed-up in a decade or more

Table 5: **PDS models—Solution times**

Instance	CPLEX 1.0	CPLEX 5.0 Dual	CPLEX 7.1 Primal	CPLEX 7.1 Dual
pds100	–	50413.1	2414.8	256.3
pds90	–	59981.0	2452.2	320.3
pds80	–	42055.4	2201.5	304.4
pds70	335292.1	21120.4	1504.1	197.8
pds60	205798.3	7442.6	852.4	160.5
pds50	122195.9	8509.9	493.2	114.6
pds40	58920.3	2816.8	188.3	79.3
pds30	15891.9	1154.9	74.8	39.1
pds20	5168.8	232.6	27.9	20.9
pds10	208.9	13.0	3.7	2.6
pds06	26.4	2.4	1.4	0.9
pds02	0.4	0.1	0.1	0.1

From: Bixby: Solving Real-World Linear Programs a decade and More of Progress

Best Simplex

Table 10: **Solution times–Best simplex**

Model	CPLEX 1.0	CPLEX 2.2	CPLEX 5.0	CPLEX 7.1	Algorithm
car	1555.0	701.1	275.8	120.6	primal
continent	364.7	110.5	104.4	46.7	primal
energy1	1217.4	275.0	260.5	22.6	dual
energy2	10130.1	736.0	664.0	693.9	dual
energy3	21797.1	271.9	229.1	161.7	dual
fuel	5619.5	1123.2	698.6	675.0	primal
initial	3832.2	102.2	51.3	15.5	dual
schedule	152404.0	252.3	220.8	64.6	dual

From: Bixby: Solving Real-World Linear Programs a decade and More of Progress

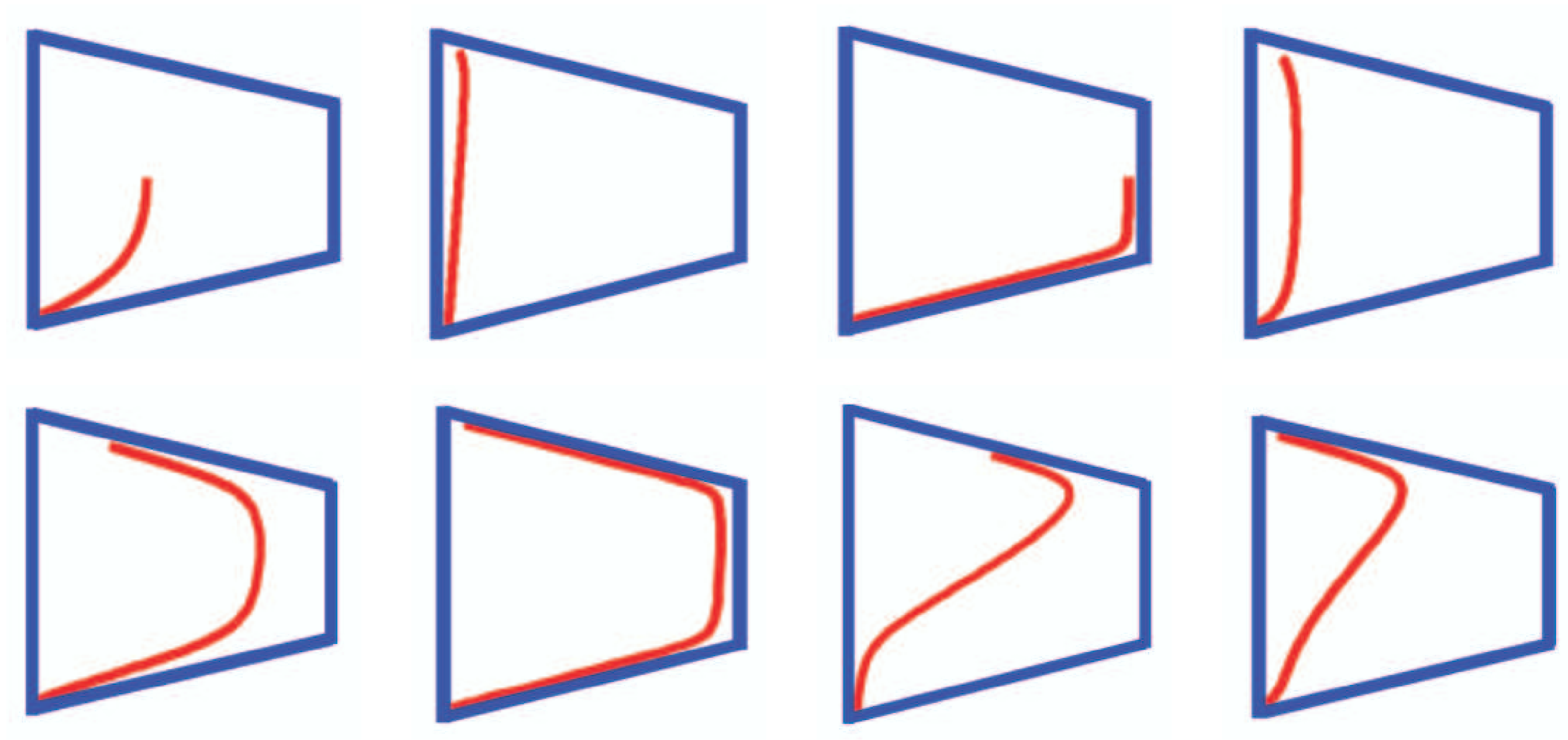
Best of Simplex–Interior Point Methods

Table 11: **Solution times–Best of three**

Model	CPLEX 1.0	CPLEX 2.2	CPLEX 5.0	CPLEX 7.1	Algorithm
car	1555.0	203.0	117.1	67.3	barrier
continent	364.7	110.5	99.5	46.7	primal
energy1	1217.4	46.5	31.5	22.4	barrier
energy2	10130.1	171.4	71.7	32.4	barrier
energy3	21797.1	152.6	113.4	82.2	barrier
fuel	5619.5	999.1	340.5	124.7	barrier
initial	3832.2	102.2	51.3	15.5	dual
schedule	152404.0	252.3	132.0	47.9	barrier

From: Bixby: Solving Real-World Linear Programs a decade and More of Progress

What next? The effects of redundancy on IPMs

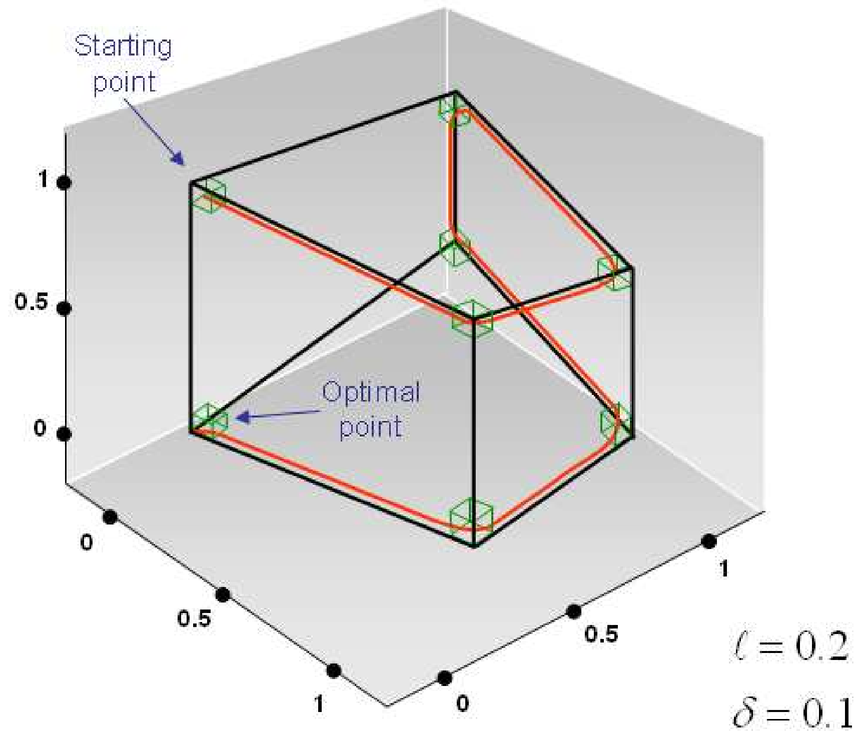


What next? The Klee-Minty Example for IPMs

How Curly Can the Central Path Be? The central path be bent along the edges of the Klee-Minty cube so that it visits a neighborhood of all 2^n vertices of that cube?

IF

we	carefully	add
an	exponential	number
of	redundant	constraints



IPM's complexity cannot be improved!

What is next? Today

- New IPM variants, closer to what is implemented (SR/PC)
- Conic IPM software – commercialize??
- Interior Point Methods for nonconvex problems, MPEC.
- Approximation algorithms for nonconvex and combinatorial optimization problems.
- New models, methodology, e.g., robust optimization, clustering, data mining.
- Non-interior, cheap first order methods for very large scale SDO problems.
- Conic Decomposition methods.
- Warm start and decomposition/cutting plane algorithms.
- New algorithms — Watch Klee-Minty!

What is next? New ideas wanted!!

- Pivot algorithms – (admissible pivot methods)
  Exponential in worst case — combinatorics, degeneracy hurts
- Ellipsoid methods – very geometrical
  Polynomial (in # of variables) — average case = worst case
- Interior Point Methods – (path following)
  Polynomial (in # of inequalities) — redundancy hurts
- Volumetric center IPMs – more geometrical (better than Ellipsoid)
  Polynomial (in # of variables) — average case = worst case
- Hyperbolic programming/Shrink-wrapping (Renegar, Zinchenko)
- New-old algorithms: Gravity method (Murty);
  Perceptron Algorithm (Dunagan, Vempala)
- New algorithms — Watch Klee-Minty!
  Distributed/parallel algorithms?? Is time ripe for it??

Conclusions

- The IPM revolution (coined by Margaret Wright) changed the landscape of optimization.
- A 10^6 efficiency improvement of software (Bob Bixby) in the last 15 years.h
- New efficiently solvable models, and thus
- a new treasure of applications were discovered.
- IPM knowledge is crystallizing, — still lot to do, we are getting to understand the weaknesses of IPMs.
- **BUT – WHAT is NEXT??**

What is next? Conclusion

I DO NOT KNOW!

Waiting for,
preparation for the next break-through.

– New Convex Optimization Problems – Cone Linear Optimization Problems

Primal-dual pair of CLO problems is given as

$$(P) \quad \min \quad c^T x \qquad\qquad (D) \quad \max \quad b^T y$$

$$\text{s.t.} \quad Ax - b \in \mathcal{C}_1 \qquad\qquad \text{s.t.} \quad c - A^T y \in \mathcal{C}_2^*$$

$$x \in \mathcal{C}_2 \qquad\qquad y \in \mathcal{C}_1^*,$$

where $b, y \in \mathbb{R}^m$, $c, x \in \mathbb{R}^n$, $A : m \times n$ matrix, $\mathcal{C}_1, \mathcal{C}_2$ are convex cones and $\mathcal{C}_i^* = \{s \in \mathbb{R}^n \ : \ x^T s \geq 0, \ \forall x \in \mathcal{C}_i\}$ are the dual cones for $i = 1, 2$.

These are solvable efficiently (in polynomial time) by using interior point methods.

LO is based on polyhedral cones.

– New Convex Optimization Problems – Second Order Conic Optimization

The second order cone in $\mathbb{R}^n$ is defined as

$$\mathcal{S}_2^n := \left\{ x \in \mathbb{R}^n : \sqrt{\sum_{i=1}^{n-1} x_i^2} \le x_n \right\}.$$

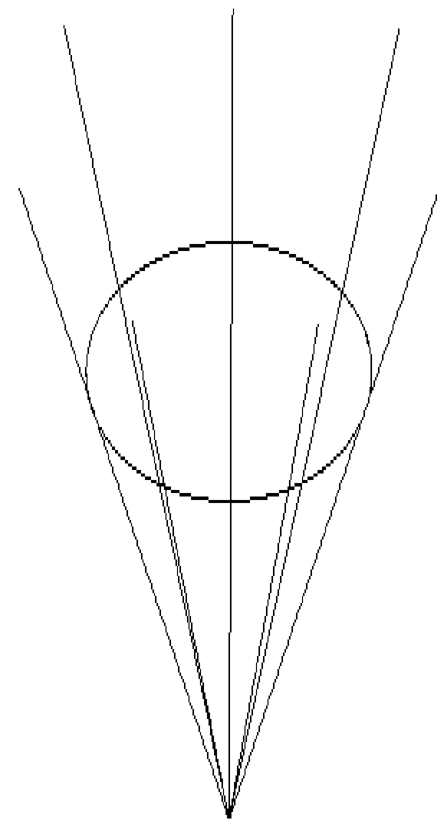
The name "ice cream cone" is coming from the 3-dimensional shape of the cone.

The second order cone is self-dual: $(\mathcal{S}_2^n)^* = \mathcal{S}_2^n$.

Optimization problems where the cones $\mathcal{C}_1$ and $\mathcal{C}_{\sqrt[n]{/}_2}$ are polyhedral and second order cones are *second order cone optimization(SOCO)* problems.

Significance

Norm minimization, robust optimization.



The ice-cream cone L^3

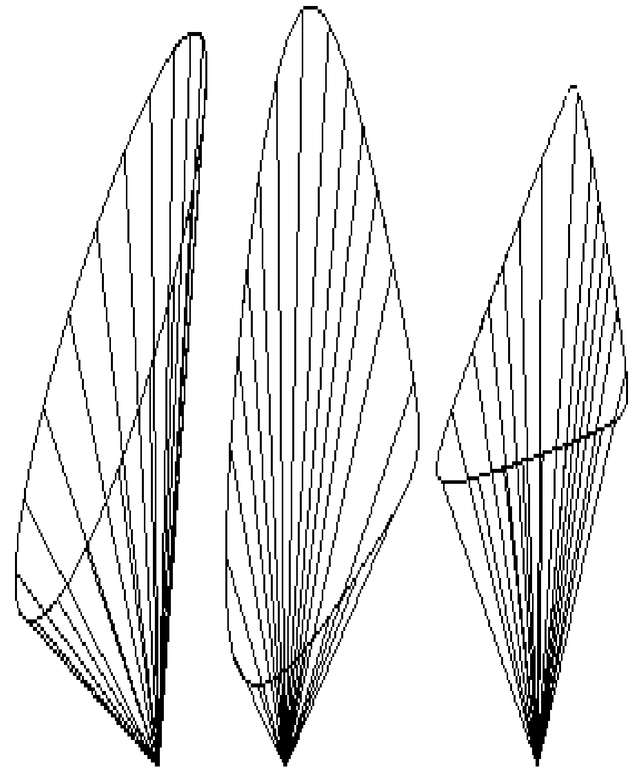
– New Convex Optimization Problems – Semidefinite Optimization – I

The semidefinite cone in $\mathbb{R}^{n\times n}$ is defined as

$$\mathcal{S}^n := \left\{X \in \mathbb{R}^{n\times n} : X = X^T, z^T X z \geq 0 \, \forall z \in \mathbb{R}^n\right\}$$

i.e. the matrices X are symmetric and positive semidefinite, denoted as $X \succeq 0$. The semidefinite cone is self-dual: $(\mathcal{S}^n)^* = \mathcal{S}^n$.

Optimization problems where the cones $\mathcal{C}_1$ and $\mathcal{C}_2$ are either polyhedral, second order or semidefinite cones are called *semidefinite optimization (SDO) problems*.



3 random 3D cross-sections of S^3_+

– New Convex Optimization Problems – Semidefinite Optimization

Let A_i, $i = 1, \cdots, n$ and C, X be $n \times n$ symmetric matrices, $b, y \in \mathbb{R}^m$ and let $\mathrm{TR}(\cdot)$ denote the trace of a matrix.

The primal-dual SDO problem is defined as

$$
\begin{array}{llll}
(SP) & \min & \mathrm{TR}(CX) & \\
 & \text{s.t.} & \mathrm{TR}(A_i X) - b_i & \geq 0, \ \forall i \\
 & & X & \succeq 0
\end{array}
\qquad
\begin{array}{llll}
(SD) & \max & b^T y & \\
 & \text{s.t.} & C - \sum_{i=1}^{m} A_i y_i & \succeq 0 \\
 & & y & \geq 0.
\end{array}
$$

Robust optimization, trust design

Linear matrix inequalities

Convex relaxation of nonconvex/integer problems

Solvability of CLO problems – Use IPMs

Classic Linear Optimization

Large scale LO problems are solved efficiently.
High performance packages, like (CPLEX, XPRESS-MP, MOSEK, McIPM) offer simplex and interior point solvers as well. Problems solved with 10^7 variables.

SOCO and SDO

Polynomial solvability established.
Traditional software is unable to handle conic constraints.
Specialized software is developed. (SeDuMi, SDPpack, SDPA, SDPT3, CSDP, SDPHA, , MOSEK etc.)
SOCO: Problems solved with 10^6 variables.
SDO: solved with 10^4 dimensional matrices.

IPMs for General Nonlinear Problems

Polynomial solvability established for convex problems.

Implementations for non-convex problems as well.

Specialized software is developed. (MOSEK, LOQO, etc.)

Problems solved with 10^4 dimensional matrices.

Robust Linear Optimization

$$(P) \quad \min \quad c^T x$$
$$\text{s.t.} \quad a_j^T x - b_j \geq 0 \qquad \forall\, j$$

Let (a_j, b_j) be uncertain, it is coming from an ellipsoid (e.g. level set of a distribution):

$$\left\{ \begin{pmatrix} a_j \\ b_j \end{pmatrix} = \begin{pmatrix} a_j^0 \\ b_j^0 \end{pmatrix} + Pu \;\middle|\; u \in \mathbb{R}^k,\ u^T u \leq 1 \right\}$$

The inequality $a_j^T x \geq b_j$ must be true for all possible values of (a_j, b_j):

$$\left[\begin{pmatrix} a_j \\ b_j \end{pmatrix} + Pu \right]^T \begin{pmatrix} x \\ 1 \end{pmatrix} \geq 0 \quad \forall\, u : u^T u \leq 1 \quad \text{iff} \quad a_j^T x - b_j + \min_{u^T u \leq 1} \left\{ (Pu)^T \begin{pmatrix} x \\ 1 \end{pmatrix} \right\} \geq 0$$

$$a_j^T x - b_j + \left\| P^T \begin{pmatrix} x \\ 1 \end{pmatrix} \right\|_2 \geq 0$$

This is a SOC (norm) constraint:

$$\left\| P^T \begin{pmatrix} x \\ 1 \end{pmatrix} \right\|_2 \leq a_j^T x - b_j.$$