Second Order Cone Programming and its Applications

Adam N. Letchford

Department of Management Science
Lancaster University Management School

Cardiff, February 2012

Outline

Introduction

We start with Linear Programming (LP).

Linear programs can be written in the form:

$$\begin{array}{rc} \min & c^T x \\ \text{s.t.} & Ax \geq b \\ & x \geq 0, \end{array}$$

where x is a vector of decision variables, A is a matrix of suitable dimension, and c and b are vectors of suitable dimension.

Introduction (cont.)

LP is well known and it enjoys the following advantages:

- A wide array of applications (statistics, economics, finance, OR...)
- An elegant theory (duality, sensitivity analysis, degeneracy...)
- Efficiently solvable (in both theory and practice).

On the other hand, the linearity assumption is restrictive.

Introduction (cont.)

People have therefore considered Non-Linear Programming (NLP).

Non-linear programs can be written in the form:

$$\begin{array}{rl} \inf & f^0(x) \\ \text{s.t.} & f^j(x) \geq 0 \quad (j = 1, \ldots, m) \\ & x \geq 0, \end{array}$$

where $f^0, \ldots, f^m$ are arbitrary functions.

However, this class of problems is too general to be useful.

Introduction (cont.)

In the 1980s, an elegant special case of NLP was proposed, that

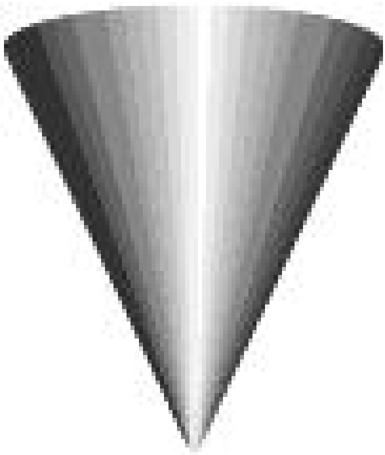
- includes LP as a special case;
- has a wide range of applications;
- can still be solved efficiently.

The purpose of this talk is to present and explain it.

As a starting point, we look at ice-cream cones!

An Ice-Cream Cone

An Idealised Ice-Cream Cone



Cones (cont.)

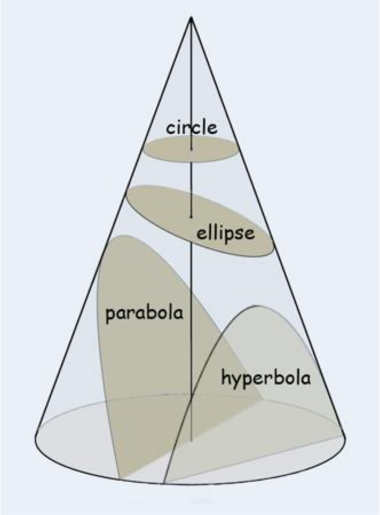
Definition (Ice-Cream Cone)

The *ice-cream cone* is the set of points in $\mathbb{R}^3$ that satisfy the inequality

$$x_3 \geq \sqrt{x_1^2 + x_2^2},$$

where x_3 is the vertical axis and x_1 and x_2 are the other two axes.

The Conic Sections (Apollonius, 200 B.C.)



Cones (Cont.)

We can generalise the ice-cream cone to higher dimensions.

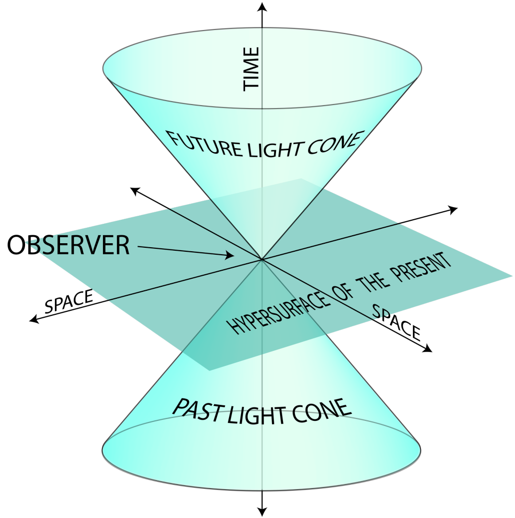
Definition (Second-Order Cone)

The *second-order cone* (SOC) is the set of points in $\mathbb{R}^n$ satisfying the inequality

$$x_n \geq \sqrt{x_1^2 + \cdots + x_{n-1}^2}.$$

The SOC is sometimes called the *Lorentz cone*, after Hendrik Lorentz, Dutch physicist (1853-1928).

The SOC in Special Relativity



Second-Order Cone Programming

Definition (SOCP)

Second-Order Cone Programming (SOCP) is a generalisation of LP in which, as well as linear constraints, there can be constraints of the form

$$x_k \geq \sqrt{\sum_{i \in S} x_i^2},$$

where k is an arbitrary variable index and S is a subset of variable indices satisfying $k \notin S$.

Second-Order Cone Programming (cont.)

Extending the idea of conic sections to higher dimension, we see that the feasible region of an SOCP can be the intersection of:

- polyhedra
- ellipsoids
- paraboloids
- hyperboloids.

Thus, SOCP generalises not only LP, but also ‘convex quadratic programming’ and ‘hyperbolic programming’.

Algorithms and Complexity

Let's recall the situation with LP:

- The *simplex method* (Dantzig, 1947) is very good in practice, but is bad in theory (exponential).
- The *ellipsoid method* (Khatchiyan, 1979) is good in theory (polynomial) but very bad in practice.
- *Interior-point methods* (Karmarkar, 1984, etc.) are very good in theory (low-order polynomial) and quite good in practice.

Algorithms and Complexity (cont.)

For SOCP the situation is as follows:

- There is no known simplex method.
- The ellipsoid method remains good in theory (Grötschel, Lovász & Schrijver, 1988), but remains very bad in practice.
- Interior-point methods remain very good in theory (Nesterov & Nemirovsky, 1988), and are fairly good in practice (though not as good as for LP).

We do however have to deal with three little technical issues...

Algorithms and Complexity (cont.)

Consider the (trivial) SOCP:

$$\min \left\{ x_3 : x_1 = 1,\, x_2 = 1,\, x_3 \geq \sqrt{x_1^2 + x_2^2} \right\}.$$

The optimal solution is $x_1^* = x_2^* = 1$, $x_3^* = \sqrt{2}$.

> *First technical issue: SOCPs can have irrational optimal solutions!*

Algorithms and Complexity (cont.)

Consider the (almost trivial) SOCP:

$$\inf \left\{ x_3 - x_2 : x_1 = 1,\, x_3 \geq \sqrt{x_1^2 + x_2^2} \right\}.$$

We can bring $x_3 - x_2$ arbitrarily close to 0 (by making x_2 arbitrarily large and setting $x_3 := \sqrt{1 + x_2^2}$), but we cannot actually reach 0.

> *Second technical issue: the infimum in an SOCP may not be attained!*

Algorithms and Complexity (cont.)

Consider the following constraints:

$$x_3 - x_2 = 1,\ x_4 - 2x_2 = 1,\ x_3 \geq \sqrt{x_1^2 + x_2^2}.$$

These can be shown to imply $x_4 \geq x_1^2$.

Chaining such constraints together we can get $x_7 \geq x_4^2$, $x_{10} \geq x_7^2$, and so on. At the end, we get $x_{3p+1} \geq x_1^{(2^p)}$.

> *Third technical issue: It can take an exponential number of bits to write the optimal solution to an SOCP, even if it is attained and rational!*

Algorithms and Complexity (cont.)

Fortunately:

- If the feasible region is *bounded*, then SOCPs can be solved to arbitrary *fixed precision* in polynomial time (Grötschel, Lovász & Schrijver, 1988).

(Ramana, 1997, showed that the SOCP *feasibility problem* is in $\mathcal{NP} \cap$ co-$\mathcal{NP}$ under the real-number model).

Applications of SOCP

SOCP has many applications in engineering, statistics, finance and operational research.

Here we mention just three applications in detail:

- constrained least-squares
- portfolio selection
- inventory control.

Application I: constrained least-squares

In statistical estimation, one encounters problems of the form:

$$\begin{array}{rc} \min & \sum_{j=1}^{m} ||y^j - x||_2^2 \\ \text{s.t.} & Ax \leq b \\ & x \in \mathbb{R}^n. \end{array}$$

The interpretation is:

- $y^1, \ldots, y^m$ are observations of a random n-vector with unknown mean;
- x is an estimate of the mean;
- $Ax \leq b$ is a set of constraints that the mean is known to satisfy;
- the goal is to minimise the sum of squared residuals.

Application I: constrained least-squares (cont.)

Introducing new variables, we re-write this as the SOCP:

$$\begin{array}{rc} \min & z \\ \text{s.t.} & z \geq \sqrt{\sum_{j=1}^{m} u_j^2} \\ & u_j \geq \sqrt{\sum_{i=1}^{n} t_{ij}^2} \quad (j = 1, \ldots, m) \\ & t_{ij} = y_i^j - x_i \qquad (i = 1, \ldots, n; j = 1, \ldots, m) \\ & Ax \leq b \\ & x \in \mathbb{R}^n \\ & z \in \mathbb{R} \\ & u \in \mathbb{R}^m \\ & t \in \mathbb{R}^{m \times n}. \end{array}$$

Application II: portfolio selection

- In the Markowitz portfolio selection model, there are n stocks for which we have a vector $r \in \mathbb{R}^n$ of *returns* and a matrix $Q \in \mathbb{R}^{n \times n}$ of covariances.
- We have to decide how much to invest in each stock.
- So we have a vector $x \in \mathbb{R}^n_+$ of decision variables.
- Our expected return is $r^T x$ and our risk (variance) is $x^T Q x$.
- Minimising risk subject to a lower bound on the return is a convex quadratic programme and therefore easily reduced to SOCP.

Application II: portfolio selection (cont.)

- A more interesting problem is to maximise return subject to a *loss risk constraint* of the form $Pr(r \leq \alpha) \leq \beta$.
- The constraint can be written as:

$$r^T x + \Phi^{-1}(\beta)\sqrt{x^T Q x} \geq \alpha,$$

  where Φ is the Normal CDF.
- Since Q is psd, we have $Q = AA^T$ for some matrix A.
- Thus the constraint can be written as:

$$r^T x + \Phi^{-1}(\beta)||Ax|| \geq \alpha.$$

- Provided $\beta < 1/2$, this can be handled by SOCP.

Application III: inventory control

The following non-linear programme represents a multi-item inventory control problem with a resource constraint (Ziegler, 1982):

$$\begin{array}{ll} \min & \sum_{i=1}^{n}(c_i x_i + d_i/x_i) \\ \text{s.t.} & \sum_{i=1}^{n} b_i x_i \le b_0 \\ & l_i \le x_i \le u_i \qquad (i = 1, \ldots, n). \end{array}$$

Kuo & Mittelmann (2004) propose to transform this formulation using the following substitution:

$$\begin{aligned} x_i &= (u_i - v_i)/2 \\ 1/x_i &= (u_i + v_i)/2. \end{aligned}$$

Application III: inventory control (cont.)

The result of the transformation is:

$$\begin{array}{lll}
\min & \sum_{i=1}^{n} \left((c_i + d_i)u_i + (d_i - c_i)v_i\right) & \\
\text{s.t.} & \sum_{i=1}^{n} b_i(u_i - v_i) \leq 2b_0 & \\
 & 2l_i \leq u_i - v_i \leq 2u_i & (i = 1, \ldots, n) \\
 & u_i \geq \sqrt{v_i^2 + 2} & (i = 1, \ldots, n) \\
 & u, v \in \mathbb{R}^n. &
\end{array}$$

This too can be solved using SOCP.

Concluding Remarks

- SOCP is an elegant generalisation of LP.
- It has a wealth of practical applications.
- It is also ‘almost’ polynomial-time solvable.
- There is now good software for SOCP.
- It is only a little slower than LP software.

Concluding Remarks (cont.)

- There is an even more general class of optimisation problems that can be solved in polynomial time: *Semidefinite Programming* or SDP.
- In SDP, one uses the cone of symmetric real positive semidefinite matrices, instead of the second-order cone.
- A huge array of applications of SDP exists.
- Unfortunately, current SDP software is rather slow.

For Further Reading

- Y. Nesterov & A. Nemirovski (1994) *Interior Point Polynomial Methods in Convex Programming: Theory and Applications*. SIAM Press: Philadelphia.
- M.G. Lobo *et al.* (1998) Applications of second-order cone programming. *Lin. Alg. Appl.*, 284, 193–228.
- A. Ben-Tal & A. Nemirovski (2001) On polyhedral approximations of the second order cone. *Math. Oper. Res.*, 26, 193-205.
- F. Alizadeh & D. Goldfarb (2003) Second-order cone programming. *Math. Program.*, 95, 3–51.