Module 05
Optimization Classes

Ahmad F. Taha

CE 5999-02 Special Topics — Intro to Optimization

Email: `ahmad.taha@vanderbilt.edu`

Webpage: `http://lab.vanderbilt.edu/taha`

VANDERBILT UNIVERSITY

January 9, 2023

In this lecture

- Linear programs
- Linear-fractional programs
- Quadratic programs
- Second-order cone programs
- Geometric programs
- Semidefinite programs

Linear program (LP)

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Gx \preceq h \\ & Ax = b \end{aligned}$$

- $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ is a vector of optimization variables
- Objective is linear; feasible set is a polyhedron
- A linear program is a convex optimization problem
- Maximizing $c^T x$ over a polyhedron is also a linear program

Standard and inequality form linear programs

Standard form LP:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array}$$

- All variables are constrained to be nonnegative. They are the only inequality constraints allowed.
- Widely used in LP software
- Every LP can be converted to standard form

Inequality form LP:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \end{array}$$

- No equality constraints

Converting LPs to standard form (1)

1. Introduce a **slack variable** $s_i \geq 0$ for each inequality constraint $(i = 1, \ldots, m)$
$$g_i^T x \leq h_i \Rightarrow g_i^T x + s_i = h_i$$
2. Express x_i as $x_i = x_i^+ - x_i^-$ with $x_i^+ \geq 0$ and $x_i^- \geq 0$

- The new optimization variables are $s = [s_1, \ldots, s_m]^T$, $x^+ = [x_1^+, \ldots, x_n^+]^T$, $x^- = [x_1^-, \ldots, x_n^-]^T$
- The objective becomes $c^T x = c^T x^+ - c^T x^-$
- One row of the inequality constraint is
$$g_i^T x + s_i = h_i \Leftrightarrow g_i^T x^+ - g_i^T x^- + s_i = h_i$$
- Organize in matrix form
$$\begin{bmatrix} g_1^T \\ \ldots \\ g_m^T \end{bmatrix} x^+ - \begin{bmatrix} g_1^T \\ \ldots \\ g_m^T \end{bmatrix} x^- + \begin{bmatrix} s_1 \\ \vdots \\ s_m \end{bmatrix} = \begin{bmatrix} h_1 \\ \vdots \\ h_m \end{bmatrix} \Leftrightarrow Gx^+ - Gx^- + s = h$$

$$\begin{array}{rl} \text{minimize} & c^T x^+ - c^T x^- \\ \text{subject to} & Gx^+ - Gx^- + s = h \\ & Ax^+ - Ax^- = b, \;\; x^+ \succeq 0, \; x^- \succeq 0, \; s \succeq 0 \end{array}$$

Converting LPs to standard form (2)

- The problem is written as an inequality form LP as follows

$$\begin{aligned} \text{minimize} \quad & c^T x^+ - c^T x^- \\ \text{subject to} \quad & Gx^+ - Gx^- + s = h \\ & Ax^+ - Ax^- = b \\ & x^+ \succeq 0,\ x^- \succeq 0,\ s \succeq 0 \end{aligned}$$

- To see it more clearly, define the variable $y = [x_1^+, \ldots, x_n^+, x_1^-, \ldots, x_n^-, s_1, \ldots, s_m]^T$

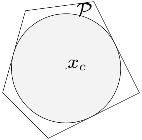
$$\begin{aligned} \text{minimize} \quad & \begin{bmatrix} c \\ -c \\ 0_{m\times 1} \end{bmatrix}^T y \\ \text{subject to} \quad & \begin{bmatrix} G & -G & I_{m\times m} \\ A & -A & 0_{p\times n} \end{bmatrix} y = \begin{bmatrix} h \\ b \end{bmatrix} \\ & y \succeq 0 \end{aligned} \quad \Leftrightarrow \quad \begin{aligned} \text{minimize} \quad & \tilde{c}^T y \\ \text{subject to} \quad & \tilde{A} y = \tilde{b} \\ & y \succeq 0 \end{aligned}$$

Chebyshev center of a polyhedron

Find largest ball inside a polyhedron

$$\mathcal{P} = \{x | a_i^T x \le b_i,\ i = 1, \dots, m\}$$

Center is called Chebyshev center of $\mathcal{P}$



Ball $\{x_c + u \,|\, \|u\|_2 \le r\}$ lies in $\mathcal{P}$ if and only if

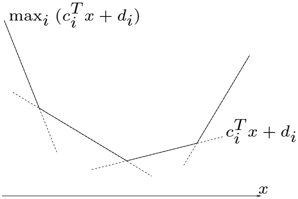
$$\max\{a_i^T x_c + a_i^T u \,|\, \|u\|_2 \le r\} \le b_i, \quad i = 1, \dots, m$$

Finding the Chebyshev center is an LP with variables $x_c \in \mathbb{R}^n$ and $r \in \mathbb{R}$

$$\begin{array}{ll} \text{maximize} & r \\ \text{subject to} & a_i^T x_c + r\|a_i\|_2 \le b_i, \quad i = 1, \dots, m \end{array}$$

Piecewise-linear minimization

$$\begin{array}{ll} \text{minimize} & \max_{i=1,\dots,K} \{c_i^T x + d_i\} \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$



is equivalent to an LP with variables $x \in \mathbb{R}^n$ and $t \in \mathbb{R}$

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & c_i^T x + d_i \le t, \; i = 1, \dots, K \\ & Gx \preceq h \\ & Ax = b \end{array}$$

Linear-fractional programming

$$\begin{array}{ll}\text{minimize} & (c^T x + d)/(f^T x + e) \\ \text{subject to} & Ax - b = 0,\ Gx \preceq h,\ f^T x + e > 0\end{array}$$

- quasiconvex optimization problem
- can be solved efficiently via bisection; each step is an LP feasibility problem
- see textbook for alternative LP formulation

Extension:

$$\begin{array}{ll}\text{minimize} & \max\limits_{i=1,\ldots,K} \{(c_i^T x + d_i)/(f_i^T x + e_i)\} \\ \text{subject to} & Ax - b = 0,\ Gx \preceq h,\ f_i^T x + e_i > 0,\ i = 1, \ldots, K\end{array}$$

Quadratic functions and quadratic forms

- quadratic function

$$f(x) = x^T P x + q^T x + r$$

convex if and only if $P \succeq 0$

- quadratic form

$$f(x) = x^T P x$$

convex if and only if $P \succeq 0$

(P is always symmetric: $P \in \mathbb{S}^n$)

Quadratic programs

Quadratic program (QP):

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

Quadratically constrained quadratic program (QCQP):

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} & (1/2)x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

- Convex problems if $P_0, P_1, \ldots, P_m$ are positive semidefinite
- Very hard to solve if nonconvex
- LP $\subset$ QP (set $P_0 = 0$) $\subset$ QCQP (set $P_i = 0$, $i = 1, \ldots, m$)

Nonconvex extensions of LP and QP

Boolean LP or **zero-one LP**:

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Gx \preceq h,\ Ax = b \\ & x_i \in \{0, 1\}, \quad i = 1, \dots, n \end{aligned}$$

mixed-integer LP (some variables take integer values, some others continuous):

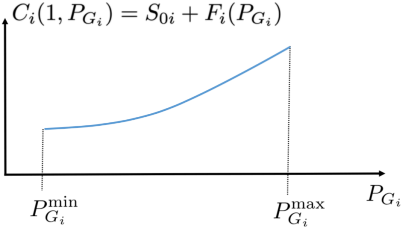
$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Gx \preceq h,\ Ax = b \\ & x_i \in \mathbb{Z},\ i = 1, \dots, \tilde{n}; \quad x_i \in \mathbb{R},\ i = \tilde{n} + 1, \dots, n \end{aligned}$$

These are in general very hard to solve.

Application: Static unit commitment

- Extend the economic dispatch to include on/off scheduling of units
- Binary variable: $u_i = 1$ if unit i will be scheduled to be on; 0 otherwise
- If unit i is scheduled to be on, it has to produce at least $P_{G_i}^{\min}$
- ... and incur nonzero *startup cost* $S_{0i} > 0$
- Cost function $C_i(u_i, P_{G_i}) = u_i S_{0i} + F_i(P_{G_i})$, where $F_i(P_{G_i})$ is convex in P_{G_i}
- Optimization problem with variables P_{G_i} and u_i, $i = 1, \ldots, n$

$$\begin{aligned}\min \quad & \sum_{i=1}^{N} C_i(u_i, P_{G_i}) \\ \text{subj. to} \quad & \sum_{i=1}^{N} P_{G_i} = P_L \\ & u_i P_{G_i}^{\min} \le P_{G_i} \le u_i P_{G_i}^{\max}, \\ & u_i \in \{0,1\}, \; i = 1, \ldots, N\end{aligned}$$



Second-order cone programs (SOCP)

$$\begin{aligned} \min \quad & f^T x \\ \text{subj. to} \quad & \|A_i x + b_i\|_2 \leq c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g \end{aligned}$$

- What is the constraint $\|A_i^T x + b_i\|_2 \leq c_i^T x + d_i$?
- We call this constraint the second order cone constraint. Why?
- Well, remember that the second order cone set $\mathcal{C}$ is given by

$$\mathcal{C}_{n+1} = \left\{ \begin{bmatrix} x \\ t \end{bmatrix} \middle| x \in \mathbb{R}^n, t \in \mathbb{R}, ||x||_2 \leq t \right\}$$

- Also known by quadratic cone, ice-cream cone, or Lorentz cone
- The second-order cone in $\mathbb{R}^3$ is $\left\{ (x, y, z) \middle| \sqrt{x^2 + y^2} \leq z \right\}$
- A generalization of this is the affine mapping $Ax + b$ or the SOC constraint
- When is an SOCP an LP? Or a QCQP?
- SOCP includes LP with $A_i = 0$, $i = 1, \ldots, m$
- SOCP includes convex QCQP with $c_i = 0$, $i = 1, \ldots, m$

Important notes

$$\begin{aligned} \min \quad & f^T x \\ \text{subj. to} \quad & \|A_i x + b_i\|_2 \leq c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g \end{aligned}$$

- The SOCP constraint can equivalently be written as a quadratic constraint

$$\|A_i x + b_i\|_2^2 - (c_i^T x + d_i)^2 \leq 0$$

- But such quadratic is *not convex in general*
- Bottomline: We need to keep the constraint in the form

$$\|A_i x + b_i\|_2 \leq c_i^T x + d_i$$

- SOCP can be written as an LMI/SDP:

$$||x|| \leq t \Leftrightarrow \begin{bmatrix} tI & x \\ x^T & t \end{bmatrix} \succcurlyeq 0$$

or more generally

$$\|A_i x + b_i\|_2 \leq c_i^T x + d_i \Leftrightarrow \begin{bmatrix} (c_i^T x + d_i)I & A_i x + b_i \\ (A_i x + b_i)^T & c_i^T x + d_i \end{bmatrix} \succcurlyeq 0$$

Facility location (Weber problem)

- Given M points $z_i \in \mathbb{R}^d$ $(i = 1, \ldots, M)$, find the location p so that the sum of distances of p from these points is minimized

$$\min \quad \sum_{i=1}^{M} \|p - z_i\|_2$$

- Using the epigraph trick: Introduce variable t_i such that $\|p - z_i\|_2 \leq t_i$
- The problem becomes SOCP with variables $t_i \in \mathbb{R}$ $(i = 1, \ldots, M)$ and $p \in \mathbb{R}^d$

$$\begin{aligned} \min \quad & \sum_{i=1}^{M} t_i \\ \text{subject to} \quad & \|p - z_i\|_2 \leq t_i, \quad i = 1, \ldots, M \end{aligned}$$

SOCP solver example

- SeDuMi is a solver for problems of the form

$$\begin{array}{ll} \min & c^T x \\ \text{subject to} & Ax = b \\ & x \in \mathcal{K} \end{array} \qquad\qquad \begin{array}{ll} \max & b^T y \\ \text{subject to} & c - A^T y \in \mathcal{K} \end{array}$$

- where $\mathcal{K}$ is a cone:
  - nonnegative orthant: $\mathbb{R}_+^n$ (then the left problem is a standard form LP and the right problem an inequality form LP)
  - the second-order (quadratic) cone: $\mathcal{Q} = \{(u, v) \in \mathbb{R}^{n+1} \,|\, u \geq \|v\|_2\}$
  - the positive semidefinite cone (see later)
- The command is `[x,y,info] = sedumi(A,b,c,K)` `http://sedumi.ie.lehigh.edu/`
- SeDuMi solves **both problems** at the same time based on the input data $A, b, c, \mathcal{K}$

Geometric programming

monomial function with $c \geq 0$, $\alpha_j \in \mathbb{R}$, $\mathrm{dom}(f) = \{x \,|\, x \succ 0\}$

$$f(x) = c x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$$

posynomial function: nonnegative sum of monomials

$$f(x) = \sum_{k=1}^{K} c_k x_1^{\alpha_{1k}} x_2^{\alpha_{2k}} \cdots x_n^{\alpha_{nk}}$$

geometric program (this is not a convex problem)

$$\begin{array}{ll} \min & f_0(x) \\ \text{subj. to} & f_i(x) \leq 1, \quad i = 1, \ldots, n \\ & h_i(x) = 1, \quad i = 1, \ldots, p \end{array}$$

with $f_0, \ldots, f_n$ posynomials and $h_1, \ldots, h_p$ monomials

GP in convex form

Change of variables: $y_i = \log x_i \Leftrightarrow x_i = e^{y_i}$
If f(x) is monomial, then $\log f(e^{y_1}, \ldots, e^{y_n})$ is *affine*

$$f(x) = cx_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n} \Rightarrow \log f(e^{y_1}, \ldots, e^{y_n}) = \alpha_1 y_1 + \ldots + \alpha_n y_n + \log c$$

If f(x) is posynomial, then $\log f(e^{y_1}, \ldots, e^{y_n})$ is *convex*

$$f(x) = \sum_{k=1}^{K} c_k x_1^{\alpha_{1k}} x_2^{\alpha_{2k}} \cdots x_n^{\alpha_{nk}} \Rightarrow \log f(e^{y_1}, \ldots, e^{y_n}) = \log \sum_{k=1}^{K} e^{\alpha_{1k} y_1 + \ldots + \alpha_{nk} y_n + \log c_k}$$

Geometric program in equivalent convex form with variables $y_1, \ldots, y_n$

$$\begin{array}{ll} \min & \log f_0(e^{y_1}, \ldots, e^{y_n}) \\ \text{subj. to} & \log f_i(e^{y_1}, \ldots, e^{y_n}) \leq 0, \quad i = 1, \ldots, n \\ & \log h_i(e^{y_1}, \ldots, e^{y_n}) = 0, \quad i = 1, \ldots, p \end{array}$$

Semidefinite programming

Semidefinite program (SDP) with variable $x \in \mathbb{R}^n$

$$\begin{aligned} \min \quad & c^T x \\ \text{subj. to} \quad & \sum_{i=1}^{n} x_i F_i + G \preceq 0 \\ & Ax = b \end{aligned}$$

where $F_1, \ldots, F_n, G \in \mathbb{S}^m$ (i.e., $F_1, \ldots, F_n, G$ are **symmetric** matrices)

- Constaint $\sum_{i=1}^{n} x_i F_i + G \preceq 0$ is called a linear matrix inequality (LMI)

Standard and inequality form SDP

Standard SDP with variable $X \in \mathbb{S}^n$ (symmetric matrix)

$$\begin{array}{ll} \min & \mathrm{tr}(CX) \\ \text{subj. to} & \mathrm{tr}(A_iX) = b_i, \quad i = 1, \ldots, m \\ & X \succeq 0 \end{array}$$

where $C, A_i \in \mathbb{S}^n$

Inequality SDP with variable $x \in \mathbb{R}^n$

$$\begin{array}{ll} \min & c^T x \\ \text{subj. to} & \sum_{i=1}^{n} x_i A_i \preceq B \end{array}$$

where $A_i, B \in \mathbb{S}^m$

LP vs SDP

General LP

$$\begin{aligned} \min\ & c^T x \\ \text{s. to } & Gx \preceq h \\ & Ax = b \end{aligned}$$

Standard LP

$$\begin{aligned} \min\ & c^T x \\ \text{s. to } & Ax = b \\ & x \succeq 0 \end{aligned}$$

Inequality LP

$$\begin{aligned} \min\ & c^T x \\ \text{s. to } & Ax \preceq b \end{aligned}$$

General SDP

$$\begin{aligned} \min\ & c^T x \\ \text{s. to } & \sum_{i=1}^{n} x_i F_i + G \preceq 0 \\ & Ax = b \end{aligned}$$

Standard SDP

$$\begin{aligned} \min\ & \operatorname{tr}(CX) \\ \text{s. to } & \operatorname{tr}(A_i X) = b_i \\ & (i = 1, \ldots, m) \\ & X \succeq 0 \end{aligned}$$

Inequality SDP

$$\begin{aligned} \min\ & c^T x \\ \text{s. to } & \sum_{i=1}^{n} x_i A_i \preceq B \end{aligned}$$

$G, F_i \in \mathbb{S}^m$ $\quad$ $C, A_i \in \mathbb{S}^n$ $\quad$ $A_i, B \in \mathbb{S}^m$

- SDP becomes LP if the respective matrices are diagonal

Multiple LMIs

Multiple LMIs

$$\sum_{i=1}^{n} x_i F_i^{(j)} + G^{(j)} \preceq 0, \quad j = 1, \dots, K$$

can be written as a single LMI

$$\begin{bmatrix} \sum_{i=1}^{n} x_i F_i^{(1)} + G^{(1)} & & \\ & \ddots & \\ & & \sum_{i=1}^{n} x_i F_i^{(1)} + G^{(K)} \end{bmatrix} \preceq 0$$

or equivalently

$$\sum_{i=1}^{n} x_i \begin{bmatrix} F_i^{(1)} & & \\ & \ddots & \\ & & F_i^{(K)} \end{bmatrix} + \begin{bmatrix} G^{(1)} & & \\ & \ddots & \\ & & G^{(K)} \end{bmatrix} \preceq 0$$

Linear inequalities

Linear inequalities $Ax \leq b$ with matrix $A = [a_1, \ldots a_n] \in \mathbb{R}^{m \times n}$ can be written as LMI

$$\sum_{i=1}^{n} x_i \operatorname{diag}(a_i) - \operatorname{diag}(b) \preceq 0$$

where $\operatorname{diag}(a_i)$ is a diagonal matrix with the entries of a_i on the diagonal.

Proof: For eny vector $v \in \mathbb{R}^m$, it holds that $v \preceq 0 \Leftrightarrow \operatorname{diag}(v) \preceq 0$.
(Notice the different meaning of $\preceq$ above.)
So we have that $Ax - b \preceq 0 \Leftrightarrow \operatorname{diag}(Ax - b) \preceq 0$ and

$$\operatorname{diag}(Ax) = \operatorname{diag}\left(\sum_{i=1}^{n} x_i a_i\right) = \sum_{i=1}^{n} \operatorname{diag}(x_i a_i) = \sum_{i=1}^{n} x_i \operatorname{diag}(a_i).$$

LMIs with linear functions of vector variable

Example: The LMI with variable $x \in \mathbb{R}^n$

$$\begin{bmatrix} a^T x + b & c^T x + d \\ c^T x + d & f^T x + g \end{bmatrix} \preceq 0$$

can be written as

$$\sum_{i=1}^{n} x_i \begin{bmatrix} a_i & c_i \\ c_i & f_i \end{bmatrix} + \begin{bmatrix} b & d \\ d & g \end{bmatrix} \preceq 0$$

LMIs with matrix variable

- Example: Given $A \in \mathbb{R}^{m\times m}$, find $P \succ 0$ such that $A^TP + PA \prec 0$
- In control theory, the previous problem amounts to solving a Lyapunov inequality
- It can be cast as an SDP feasibility problem
- First, we convert strict inequalities to "greater than or equal," i.e., $P \succeq \epsilon_1 I$ and $A^TP + PA \preceq \epsilon_2 I$, where $\epsilon_1 > 0$ and $\epsilon_2 > 0$ are chosen by us and are small

$$\begin{bmatrix} -P & 0 \\ 0 & A^TP+PA \end{bmatrix} + \begin{bmatrix} \epsilon_1 I & 0 \\ 0 & -\epsilon_2 I \end{bmatrix} \preceq 0$$

- How can we convert to LMI of the form $\sum_{i=1}^n x_i F_i + G \preceq 0$?

- P is symmetric, so it has $n = \frac{m(m+1)}{2}$ free entries

$$P = \begin{bmatrix} x_1 & x_2 & \dots & x_m \\ x_2 & x_{m+1} & \dots & x_{2m-1} \\ \vdots & & \vdots & \\ x_m & x_{2m-1} & \dots & x_n \end{bmatrix}$$

- Consider a basis for the set $\mathbb{S}^m$

$$E_1 = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & & & \vdots \\ 0 & & & 0 \end{bmatrix}, \; E_2 = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & & & \vdots \\ 0 & & & 0 \end{bmatrix}, \dots, \; E_n = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & & & \vdots \\ 0 & & & 1 \end{bmatrix}$$

- The problem is equivalent to the following SDP

$$\sum_{i=1}^{n} x_i \begin{bmatrix} -E_i & 0 \\ 0 & A^T E_i + E_i A \end{bmatrix} + \begin{bmatrix} \epsilon_1 I & 0 \\ 0 & -\epsilon_2 I \end{bmatrix} \preceq 0$$

Bottomline: SDPs must contain linear forms

A problem is an SDP if it includes any combination of the following:

- An objective linear in a vector variable, e.g., $c^T x$
- An objective linear in a matrix variable, that is, $\mathrm{Tr}(BY)$ (where B and Y are symmetric; and Y is positive or negative semidefinite)
- Multiple LMIs of the form $\sum_i^N x_i F_i + G \preceq 0$
- Linear inequalities and equalities
- LMIs with $f^T x$ as matrix entry
- LMIs that include a matrix variable (e.g., $A^T P + PA \preceq -\epsilon I$)
- Linear equalities or linear inequalities that involve the trace of a matrix variable

Convex optimization modeling languages such as CVX or YALMIP accept any combination of the previous objectives or constraints.

Schur complement

Consider the symmetric block matrix $X \in \mathbb{S}^n$ (so $A = A^T$, $C = C^T$)

$$X = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix}$$

- Suppose A is invertible
  $S = C - B^T A^{-1} B$ is the **Schur complement of A in X**
- Then the following hold:
  1. $X \succ 0 \iff A \succ 0$ and $S \succ 0$
  2. If $A \succ 0$, then $X \succeq 0 \Leftrightarrow S \succeq 0$
- Likewise when C is invertible
  $S = A - BC^{-1}B^T$ is the **Schur complement of C in X**

SOCP as SDP

$$\begin{aligned} \min \quad & f^T x \\ \text{subj. to} \quad & \|A_i x + b_i\|_2 \le c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g \end{aligned}$$

- We can write the SOCP constraint as LMI using the Schur complement

$$\begin{aligned} \|A_i x + b_i\|_2 \le c_i^T x + d_i &\Leftrightarrow -(A_i x + b_i)^T (A_i x + b_i) + (c_i^T x + d_i)^2 \ge 0 \\ &\Leftrightarrow (c_i^T x + d_i) - (A_i x + b)^T \left(\frac{1}{c_i^T x + d_i} I \right) (A_i x + b_i) \ge 0 \end{aligned}$$

Using the Shur complement, we obtain an SDP

$$\begin{aligned} \min \quad & f^T x \\ \text{subj. to} \quad & \begin{bmatrix} (c_i^T x + d_i) I & A_i x + b_i \\ (A_i x + b_i)^T & c_i^T x + d_i \end{bmatrix} \succeq 0, \quad i = 1, \ldots, m \\ & Fx = g \end{aligned}$$

Maximum eigenvalue minimization

$$\min_{x\in\mathbb{R}^n} \lambda_{\max}(A(x))$$

where $A(x) = A_0 + x_1A_1 + \ldots + x_nA_n$ and $A_i \in \mathbb{S}^k$ [so $A(x)$ is a symmetric matrix]
The problem is equivalent to SDP with variables $x \in \mathbb{R}^n$ and $t \in \mathbb{R}$:

$$\begin{aligned} \text{minimize} \quad & t \\ \text{subj. to} \quad & A(x) \preceq tI \end{aligned}$$

Proof:

$$\lambda_{\max}(A) \leq t \iff \lambda_{\max}(A - tI) \leq 0 \iff A - tI \preceq 0$$

where we used the fact that $\lambda_i(A + tI) = \lambda_i(A) + t$, $i = 1, \ldots, n$

Spectral norm minimization

Spectral norm of $A \in \mathbb{R}^{p\times q}$: $\|A\|_2 = \sigma_{\max}(A) = \sqrt{\lambda_{\max}(A^TA)}$

$$\min_{x\in\mathbb{R}^n} \sqrt{\lambda_{\max}(A(x)^TA(x))}$$

where $A(x) = A_0 + x_1A_1 + \ldots + x_nA_n$ and $A_i \in \mathbb{R}^{p\times q}$ [so $A(x)$ is rectangular]
The problem is equivalent to SDP with variables $x \in \mathbb{R}^n$ and $t \in \mathbb{R}$:

$$\begin{aligned} \text{minimize} \quad & t \\ \text{subj. to} \quad & \begin{bmatrix} tI & A(x) \\ A(x) & tI \end{bmatrix} \succeq 0 \end{aligned}$$

Proof: Using the Schur complement

$$\sqrt{\lambda_{\max}(A^TA)} \leq t \Longleftrightarrow A^TA \leq t^2I, t \geq 0 \Longleftrightarrow \begin{bmatrix} tI & A \\ A^T & tI \end{bmatrix} \succeq 0$$

Hierarchy of convex optimization problems

LP $\subset$ (convex) QP $\subset$ (convex) QCQP $\subset$ SOCP $\subset$ SDP

- SDP includes SOCP
- As SOCP includes QCQP, QP, and LP, we conclude that SDP is the most general class
- GP is outside of this hierarchy
- Recall that there are also nonconvex QPs and QCQPs (very hard to solve in general)
- A more restricted problem class is more computationally efficient (faster to solve)
    - Better to have LP with a million variables than SDP with a million variables

Solvers

- Solver: Function to solve LP, QP, SOCP, SDP, given problem data
- You must bring the problem to standard form or in general, a form that the solver accepts
- Examples: MATLAB's linprog, quadprog
- SeDuMi (`http://sedumi.ie.lehigh.edu/`)
- SDPT3 (`http://www.math.nus.edu.sg/~mattohkc/sdpt3.html`)
- MOSEK (`https://www.mosek.com/`)
- GUROBI (`https://www.gurobi.com/`)
- CPLEX (`https://www.ibm.com/analytics/cplex-optimizer`)
- The previous software packages focus on convex programs, but some also have nonconvex QP and mixed-integer capabilities

Modeling languages

- Modeling languages: Accept the problem in a format similar to its mathematical form (minimize/subject to)
- Then convert the problem to a format acceptable for a solver, and call the solver
- Easier and shorter code, but requires extra time for parsing of the problem
- CVX (`http://cvxr.com/cvx/`)
- YALMIP (`https://yalmip.github.io/`)
    - See all solvers that YALMIP can call: `https://yalmip.github.io/allsolvers/`
- GAMS (`https://www.gams.com/`)
- AIMMS, AMPL, CPLEX Optimization Studio, Google OR-Tools

Questions And Suggestions?

Thank You!

Please visit
https://lab.vanderbilt.edu/taha/
IFF you want to know more ☺