Network Optimization
Chapter 2: Mathematical Tools

Yiyin Wang
yiyinwang@sjtu.edu.cn
Department of Automation
Shanghai Jiao Tong University

Oct. 24, 2014

- Acknowledgements

  Slides borrow heavily from lectures by Stephen Boyd (Stanford), Lieven Vandenberghe (UCLA), and Mung Chiang (Princeton)
- References
  - Z.-Q. Luo, W. Yu, "An introduction to convex optimization for communications and signal processing", *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1426-1438, Aug. 2006.
  - Z.-Q. Luo, W.-K. Ma, A. M.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems", *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20-34, May 2010

Part 2: Applications

- Linear programming (LP)
- Semidefinite programming (SDP)

Linear programming (LP)

Minimize linear function over linear inequality and equality constraints:

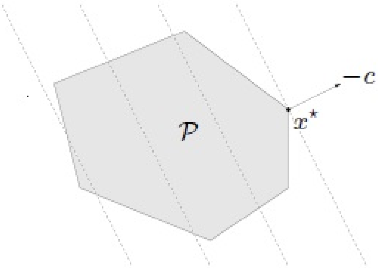
$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

where $x \in \mathbf{R}^n$, $G \in \mathbf{R}^{m \times n}$ and $A \in \mathbf{R}^{p \times n}$

- A convex problem whose feasible set is a polyhedron
- Most well-known, widely-used and efficiently-solvable optimization

Standard form LP

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



Transformation to standard form LP

1. Introduce slack variables s_i for inequality constraints:

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

2. Express x as difference between two nonnegative variables $x^+, x^- \succeq 0 : x = x^+ - x^-$

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

Now in LP standard form with variables x^+, x^-, s

Dual LP

1. Primal problem in standard form:

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

2. Write down Lagrangian using Lagrange multipliers λ, ν

$$L(x, \lambda, \nu) = c^T x - \sum_{i=1}^{n} \lambda_i x_i + \nu^T (Ax - b) = -b^T \nu + (c + A^T \nu - \lambda)^T x$$

3. Find Lagrange dual function:

$$g(\lambda, \nu) = \inf_x L(x, \lambda, \nu) = -b^T \nu + \inf_x [(c + A^T \nu - \lambda)^T x]$$

Since a linear function is bounded below only if it is identically zero, we have

$$g(\lambda, \nu) = \begin{cases} -b^T \nu & A^T \nu - \lambda + c = 0 \\ -\infty & \text{otherwise} \end{cases}$$

Dual LP

4. Write down Lagrange dual problem:

$$\begin{array}{ll} \text{maximize} & g(\lambda, \nu) = \begin{cases} -b^T\nu & A^T\nu - \lambda + c = 0 \\ -\infty & \text{otherwise} \end{cases} \\ \text{subject to} & \lambda \succeq 0 \end{array}$$

5. Make equality constraints explicit:

$$\begin{array}{ll} \text{maximize} & -b^T\nu \\ \text{subject to} & A^T\nu - \lambda + c = 0 \\ & \lambda \succeq 0 \end{array}$$

6. Simplify Lagrange dual problem:

$$\begin{array}{ll} \text{maximize} & -b^T\nu \\ \text{subject to} & A^T\nu + c \succeq 0 \end{array}$$

which is an inequality constrained LP

Network flow (graph theory notation)

- $G = (V, E)$: directed graph with vertex set V and edge set E
- b_i: external supply to each node $i \in V$
- u_{ij}: capacity of each edge $(i, j) \in E$
- c_{ij}: cost per unit flow on edge $(i, j) \in E$
- $I(i) = \{j \in V \mid (j, i) \in E\}$: set of start nodes of incoming edges to i
- $O(i) = \{j \in V \mid (i, j) \in E\}$: set of end nodes of outgoing edges from i
- Sources: $\{i \mid b_i > 0\}$
- Sinks: $\{i \mid b_i < 0\}$

Feasible flow f:

- Flow conservation: $b_i + \sum_{j \in I(i)} f_{ji} = \sum_{j \in O(i)} f_{ij}, \forall i \in V$
- Capacity constraint: $0 \leq f_{ij} \leq u_{ij}$

Network flow problem (basic formulation)

$$
\begin{aligned}
\text{minimize} \quad & \sum_{i,j \in E} c_{ij} f_{ij} \\
\text{subject to} \quad & b_i + \sum_{j \in I(i)} f_{ji} = \sum_{j \in O(i)} f_{ij}, \forall i \in V \\
& 0 \leq f_{ij} \leq u_{ij}
\end{aligned}
$$

In matrix notation as a LP:

$$
\begin{aligned}
\text{minimize} \quad & c^T f \\
\text{subject to} \quad & Af = b \\
& 0 \preceq f \preceq u
\end{aligned}
$$

where $A \in \mathbf{R}^{|V| \times |E|}$ is defined as

$$
A_{ik} = \begin{cases} 1, & i \text{ is the start node of edge } k \\ -1, & i \text{ is the end node of edge } k \\ 0, & \text{otherwise} \end{cases}
$$

A special case: maximum flow problem

The LP formulation

$$\begin{aligned} \text{maximize} \quad & b_s \\ \text{subject to} \quad & Af = b \\ & b_t = -b_s \\ & b_i = 0, \forall i \neq s, t \\ & 0 \preceq f \preceq u \end{aligned}$$

Reformulated as network flow problem:

- Costs for all edges are zero, $c_{ij} = 0, \forall i, j$
- Introduce a new edge (t, s) with infinite capacity and cost -1
- Minimize total cost is equivalent to maximize f_{ts}

Ford Fulkerson algorithm

1. Start with feasible flow f
2. Search for an augmenting path P
3. Terminate if no augmenting path
4. Otherwise, if flow can be pushed, push $\delta(P)$ units of flow along P and repeat Step 2
5. Otherwise, terminate

Augmenting path

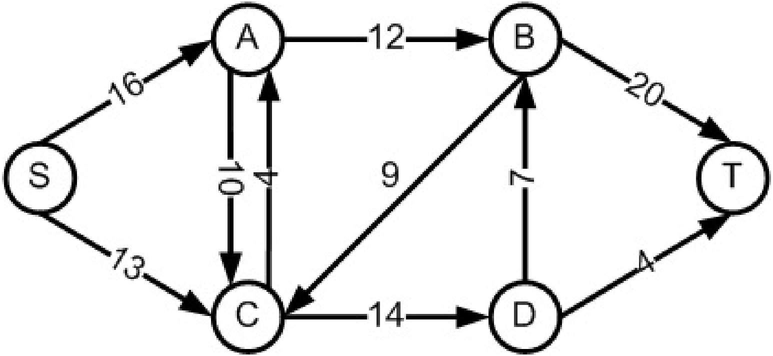
- Fine a path where we can increase flow along every forward edge and decrease flow along backward edge by the same amount. Still satisfy constraints. Increase objective function
- Augmenting path: a path from s to t such that $f_{ij} < u_{ij}$ on forward edges and f_{ij} on backward edges
  Augmenting flow amount along augmenting path P:

  $$\delta(P) = \min\{\min_{(i,j)\in F}(u_{ij} - f_{ij}), \min_{(i,j)\in B} f_{ij}\}$$

- Can search for augmenting path by following possible paths leading from s and checking conditions above

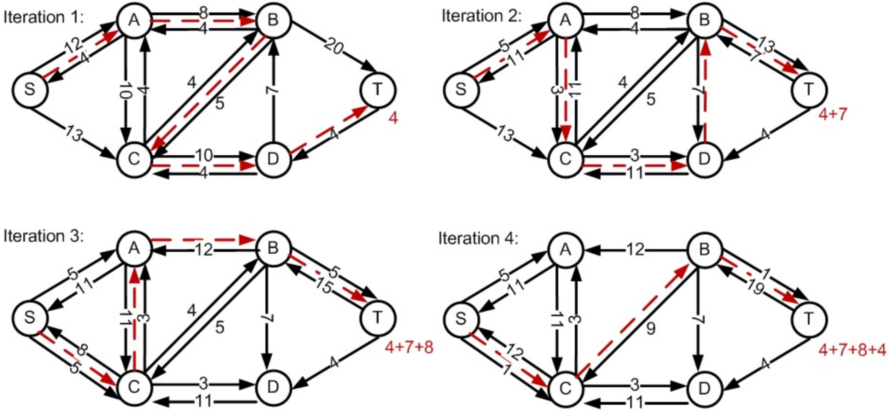
Example

What's the maximum flow?



Example

Ford Fulkerson algorithm



Linear fractional programming

Minimize ratio of affine functions over polyhedron

$$\begin{array}{ll} \text{minimize} & \dfrac{c^T x + d}{e^T x + f} \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

Domain of objective function: $\{x \mid e^T x + f > 0\}$
Not an LP. But if nonempty feasible set, transformation into an equivalent LP with variables y, z:

$$\begin{array}{ll} \text{minimize} & c^T y + dz \\ \text{subject to} & Gy - hz \preceq 0 \\ & Ay - bz = 0 \\ & e^T y + fz = 1 \\ & z \succeq 0 \end{array}$$

Let $y = \frac{x}{e^T x + f}$ and $z = \frac{1}{e^T x + f}$

Norm approximation

$$\text{minimize } \|Ax - b\|$$

where $A \in \mathbf{R}^{m \times n}$ with $m \geq n$, $\|\cdot\|$ is a norm on $\mathbf{R}^m$

- ℓ_1 norm: $\|x\|_1 = \sum_{i=1}^n |x_i|$. The norm approximation is equivalent to the LP:

$$\begin{array}{ll} \text{minimize} & \mathbf{1}^T s \\ \text{subject to} & -s \preceq Ax - b \preceq s \end{array}$$

- ℓ_∞ norm: $\|x\|_\infty = \max_i\{|x_i|\}$. The norm approximation is equivalent to the LP:

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & -t\mathbf{1} \preceq Ax - b \preceq t\mathbf{1} \end{array}$$

Part 2: Applications

- Linear programming (LP)
- Semidefinite programming (SDP)

Semidefinite programming (SDP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x_1 F_1 + x_2 F_2 + \cdots + x_n F_n + G \preceq 0 \\ & Ax = b \end{array}$$

with $F_i, G \in \mathbf{S}^k$

- Inequality constraint is called linear matrix inequality (LMI)
- Includes problems with multiple LMI constraints: for example,

$$x_1 \hat{F}_1 + x_2 \hat{F}_2 + \cdots + x_n \hat{F}_n + \hat{G} \preceq 0, \quad x_1 \tilde{F}_1 + x_2 \tilde{F}_2 + \cdots + x_n \tilde{F}_n + \tilde{G} \preceq 0$$

is equivalent to single LMI

$$x_1 \begin{bmatrix} \hat{F}_1 & 0 \\ 0 & \tilde{F}_1 \end{bmatrix} + x_2 \begin{bmatrix} \hat{F}_2 & 0 \\ 0 & \tilde{F}_2 \end{bmatrix} + \cdots + x_n \begin{bmatrix} \hat{F}_n & 0 \\ 0 & \tilde{F}_n \end{bmatrix} + \begin{bmatrix} \hat{G} & 0 \\ 0 & \tilde{G} \end{bmatrix} \preceq 0$$

Standard and inequality form SDP

Recall that $\mathrm{tr}(CX) = \sum_{i,j=1}^{n} C_{ij}X_{ij}$ is the form of a general real-valued linear function on $\mathbf{S}^n$

$$\begin{array}{ll} \text{minimize} & \mathrm{tr}(CX) \\ \text{subject to} & \mathrm{tr}(A_iX) = b_i, i = 1, \dots, p \\ & X \succeq 0 \end{array}$$

where $C, A_1, \dots, A_p \in \mathbf{S}^n$.

Dual SDP

1. Primal problem in general form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x_1 F_1 + x_2 F_2 + \cdots + x_n F_n + G \preceq 0 \\ & Ax = b \end{array}$$

   with $F_i, G \in \mathbf{S}^k$. Here f_1 is affine, and K_1 is $\mathbf{S}^k_+$

2. Write down Lagrangian using Lagrange multiplier $Z \in \mathbf{S}^n$

$$\begin{aligned} & L(x, Z, \nu) \\ = & \, c^T x + \mathrm{tr}((x_1 F_1 + x_2 F_2 + \cdots + x_n F_n + G)Z) + \nu^T (Ax - b) \\ = & \, x_1(c_1 + \mathrm{tr}(F_1 Z) + \nu^T a_1) + \cdots + x_n(c_n + \mathrm{tr}(F_n Z) + \nu^T a_n) \\ & + \mathrm{tr}(GZ) - \nu^T b \end{aligned}$$

   where a_i is the column vector of A. Note that $L(x, Z, \nu)$ is affine in x

Dual SDP

3. Find Lagrange dual function:

$$
\begin{aligned}
g(Z,\nu) &= \inf_x L(x, Z, \nu) \\
&= \begin{cases} \operatorname{tr}(GZ) - \nu^T b & c_i + \operatorname{tr}(F_i Z) + \nu^T a_i = 0 \\ -\infty & \text{otherwise} \end{cases}
\end{aligned}
$$

4. Write down Lagrange dual problem:

$$
\begin{array}{ll}
\text{maximize} & \operatorname{tr}(GZ) - \nu^T b \\
\text{subject to} & c_i + \operatorname{tr}(F_i Z) + \nu^T a_i = 0, i = 1, \ldots, n \\
& Z \succeq 0
\end{array}
$$

Strong duality obtains if the semidefinite program is strictly feasible, i.e., there exists an x with

$$
x_1 F_1 + x_2 F_2 + \cdots + x_n F_n + G \prec 0
$$

LP as SDP

LP and equivalent SDP

$$\begin{aligned} \text{LP}: \quad \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Ax \preceq b \\ \text{SDP}: \quad \text{minimize} \quad & c^T x \\ \text{subject to} \quad & \operatorname{diag}(Ax - b) \preceq 0 \end{aligned}$$

Note different interpretation of generalized inequality $\preceq$

SOCP as SDP

Second order cone programming (SOCP) and equivalent SDP

$$
\begin{array}{rll}
\text{SOCP}: & \text{minimize} & f^T x \\
& \text{subject to} & \|A_i x + b_i\|_2 \le c_i^T x + d_i, i = 1, \ldots, m \\
\text{SDP}: & \text{minimize} & f^T x \\
& \text{subject to} & \begin{bmatrix} (c_i^T x + d_i) I & A_i x + b_i \\ (A_i x + b_i)^T & c_i^T x + d_i \end{bmatrix} \succeq 0, \;\; i = 1, \ldots, m
\end{array}
$$

Matrix norm minimization

$$\text{minimize} \|A(x)\|_2 = (\lambda_{\max}(A(x)^T A(x)))^{1/2}$$

where $A(x) = A_0 + x_1 A_1 + \cdots + x_n A_n$ (with given $A_i \in \mathbf{R}^{p \times q}$)

Equivalent SDP

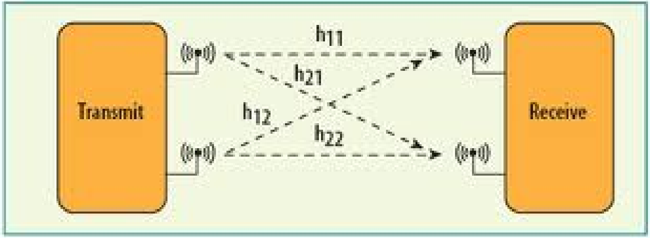
$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \begin{bmatrix} tI & A(x) \\ A(x)^T & tI \end{bmatrix} \succeq 0 \end{array}$$

- Variables $x \in \mathbf{R}^n$, $t \in \mathbf{R}$
- Constraint follows from

$$\begin{aligned} \|A\|_2 \le t \quad & \Longleftrightarrow \quad A^T A \preceq t^2 I, t \ge 0 \\ & \Longleftrightarrow \quad \begin{bmatrix} tI & A(x) \\ A(x)^T & tI \end{bmatrix} \succeq 0 \end{aligned}$$

Multi-user detection

Consider a detection and estimation problem in the receiver processing of a MIMO channel



1. This block diagram shows a typical 2 x 2 multiple-input, multiple-output (MIMO) antenna system.

Multi-user detection

The system model is given by

$$y = \sqrt{\rho/n} H s + z$$

- ρ is the (normalized) average SNR at each receive antenna
- $H \in \mathbf{C}^{n \times m}$ denotes the (known) channel matrix
- $y \in \mathbf{C}^{n}$ is the received channel output
- s is the transmitted information binary symbol vector, $s \in \{-1, 1\}^{m}$
- z is the additive white Gaussian noise with unit variance

The capacity of this MIMO channel is known to be proportional to the number of transmit antennas.

Multi-user detection

The MIMO maximum-likelihood detection:

$$\mathrm{minimize}_{s\in\{1,-1\}^m}\|y-\sqrt{\rho/n}Hs\|^2$$

- For simplicity, consider the case $m=n$ and H, s and z are real.
- Let us rewrite the log-likelihood function as

$$\|y-\sqrt{\rho/n}Hs\|^2=\mathrm{tr}(Qxx^T)$$

where $Q\in\mathbf{R}^{(n+1)\times(n+1)}$ and vector $x\in\mathbf{R}^{n+1}$ are defined as

$$Q=\begin{bmatrix}(\rho/n)H^TH & -\sqrt{\rho/n}H^Ty\\ -\sqrt{\rho/n}y^TH & \|y\|^2\end{bmatrix}, x=\begin{bmatrix}s\\ 1\end{bmatrix}$$

Multi-user detection

- Let $X = xx^T$ and note that $X \succeq 0$, $X_{ii} = 1$ and $\mathrm{rank}(X) = 1$ if and only if $X = xx^T$ for some x with $x_i = \pm 1$.
- The ML detections is equivalent to

$$\begin{array}{ll} \text{minimize} & \mathrm{tr}(QX) \\ \text{subject to} & X_{ii} = 1, i = 1, \ldots, n \\ & X \succeq 0, \mathrm{rank}(X) = 1 \end{array}$$

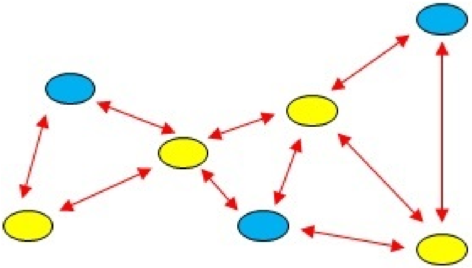
- Relaxing the rank-1 constraint, we arrive at SDP relaxation for the original problem

$$\begin{array}{ll} \text{minimize} & \mathrm{tr}(QX) \\ \text{subject to} & X \succeq 0, X_{ii} = 1, i = 1, \ldots, n \end{array}$$

Multi-user detection

- Factorization and randomization methods can be used to recover a feasible rank-1 solution from the optimal solution of the SDP relaxation
- The exhaustive search of the ML detector: NP-hard $\mathcal{O}(2^n)$
- The SDP relaxation: polynomial time $\mathcal{O}(n^{3.5})$ thank to the diagonal structure of the constraints. Its performance is close to the ML detector.

Sensor network localization (SNL)



- Yellow nodes are sensors $(x_1, \ldots, x_n)$, whose positions are to be estimated
- Blue nodes are anchors $(a_1, \ldots, a_m)$, whose positions are known
- Let E_{ss} and E_{sa} be the sets of sensor-sensor and sensor-anchor edges, respectively
- Suppose the measured distances $\{d_{ik} : (i, k) \in E_{ss}\}$ and $\{\bar{d}_{ik} : (i, k) \in E_{sa}\}$ are noise free

SNL

The SNL problem is to find $x_1, \ldots, x_n \in \mathbf{R}^2$ such that

$$P1 : \begin{array}{l} \|x_i - x_k\|^2 = d_{ik}^2, (i,k) \in E_{ss} \\ \|a_i - x_k\|^2 = \bar{d}_{ik}^2, (i,k) \in E_{sa} \end{array}$$

The problem of determining the feasibility of $P1$ is NP-hard. However, we can derive a SDP relaxation of $P1$.

SNL

- Observe that

$$\|x_i - x_k\|^2 = x_i^T x_i - 2x_i^T x_k + x_k^T x_k$$

- Let us define $X = [x_1 \ x_2 \ \ldots, x_n]$. Hence, we may write

$$\|x_i - x_k\|^2 = (e_i - e_k)^T X^T X(e_i - e_k) = \mathrm{tr}(E_{ik} X^T X)$$

where $e_i \in \mathbf{R}^n$ is the ith unit vector, and
$E_{ik} = (e_i - e_k)(e_i - e_k)^T \in \mathbf{S}^n$

SNL

- In a similar way $\|a_i - x_k\|^2 = a_i^T a_i - 2a_i^T x_k + x_k^T x_k$
- We can achieve

$$\begin{aligned} \|a_i - x_k\|^2 &= [a_i^T \;\; e_k^T] \left[\begin{array}{cc} I_2 & X \\ X^T & X^T X \end{array} \right] \left[\begin{array}{c} a_i \\ e_k \end{array} \right] \\ &= \mathrm{tr}(\bar{M}_{ik} Z) \end{aligned}$$

where

$$\bar{M}_{ik} = \left[\begin{array}{c} a_i \\ e_k \end{array} \right] [a_i^T \;\; e_k^T]$$

and

$$Z = \left[\begin{array}{cc} I_2 & X \\ X^T & X^T X \end{array} \right] = \left[\begin{array}{c} I_2 \\ X^T \end{array} \right] [I_2 \;\; X]$$

Observe that $Z \in \mathbf{S}^{n+2}$ is a rank-2 positive semidefinite matrix whose upper left 2×2 block is constrained to be an identity matrix.

SNL

- Let us define

$$M_{ik} = \begin{bmatrix} 0 & 0 \\ 0 & E_{ik} \end{bmatrix}$$

- $P1$ is equivalent to the following rank constrained SDP

$$\begin{aligned} P2 : \text{find} \quad & Z \\ \text{subject to} \quad & \operatorname{tr}(M_{ik} Z) = d_{ik}^2, (i, k) \in E_{ss} \\ & \operatorname{tr}(\bar{M}_{ik} Z) = \bar{d}_{ik}^2, (i, k) \in E_{sa} \\ & Z \succeq 0, \operatorname{rank}(Z) = 2 \\ & Z_{1:2,1:2} = I_2 \end{aligned}$$

- By dropping the rank constraint from $P2$, we can obtain an SDP relaxation of $P1$