Recap: Linear Programming

$$\text{maximize} \quad c^T x$$
$$\text{s.t.} \quad Ax \le b$$

Semi-definite matrix matrix A is a semi-definite if all eigen values are positive.

Semi-definite programming

input, output

$$\text{maximize} \quad \langle C, X \rangle \qquad \text{when} \quad \langle C, X \rangle = \sum_{i,j} \underbrace{c_{i,j} x_{i,j}}$$

→ element-wise multiplication

$n \times n$ matrix

$$\text{such that} \quad \langle A_k, X \rangle \le b_k \qquad \text{for all } k$$

input $n \times n$ matrix

and X is a semi-definite matrix

Simplified version

$$\text{maximize} \quad c^T x$$
$$\text{s.t.} \quad Ax \le b$$

→ linear program (LP)

and X is a semi-definite matrix

→ collection of variables in X or some

$\Downarrow$

$X_1, \ldots, X_p$ are semi-definite

Example maximize $\boxed{\dfrac{(c^T x)^2}{d^T x}} \Rightarrow \dfrac{(c_1 x_1 + \ldots + c_n x_n)^2}{d_1 x_1 + \ldots + d_n x_n}$ → not a linear function

when $Ax \le b$

$p = c^T x \qquad p \le c^T x, \; p \ge c^T x$

$q = d^T x \qquad p - c^T x \le 0, \; c^T x - p \le 0$

maximize t

when $Ax \le b$

and $t \le \dfrac{(c^T x)^2}{d^T x}$

$\Downarrow$

$$\boxed{\begin{array}{l} \text{maximize} \quad t \\ \text{when} \quad Ax \le b \\ \qquad p - c^T x \le 0 \\ \qquad c^T x - p \le 0 \\ \qquad q - d^T x \le 0 \\ \qquad d^T x - q \le 0 \end{array}} \quad \text{LP}$$

$t \le p^2/q$

$$D = \begin{pmatrix} p & t \\ q & p \end{pmatrix} \qquad \det(D) = p^2 - qt$$

D is semi-definite $\Leftrightarrow p^2 - qt \ge 0$

$\Leftrightarrow p^2 \ge qt$

$\Leftrightarrow \dfrac{p^2}{q} \ge t$

maximize t

such that $Ax \leq b$

$p - c^T x \leq 0$

$c^T x - p \leq 0$

$q - d^T x \leq 0$

$d^T x - q \leq 0$

$\begin{pmatrix} p & t \\ q & p \end{pmatrix}$ is semi-definite.

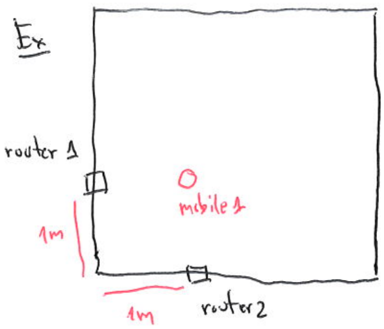
Semi-definite programming (SDP)

Can use library such as Mosek or CVX-OPT to solve.

Localization

- We want to find positions of our mobiles indoor.
- We have signal strengths between 2 mobiles
  ↳ approximated distance.
- We have signal strengths between mobiles and routers.

Ex



Mobile 1: Approximated distance to router 1 is 1 m.
Approximated distance to router 2 is 1 m

Mobile 2: Approximated distance to router 1 is 2 m

Approximated distance to router 2 is 2 m

Approximated distance to mobile 1 is 1.5 m

Bonus Question: Where Mobile 2 should be?

Goal: Find position (x, y) that minimize $(\underbrace{d[(x,y), \text{router}1] - 2}_{\text{errors in distance}})^2 + (d[(x,y), \text{router}2] - 2)^2 + (d[(x,y), \text{mobile}1] - 1.5)^2$

Localization Problem

Input: $p_1, \dots, p_m$: positions of routers [m : # routers]

$d_{i,j}$: approximate distance between router i and mobile j (for all $1 \leq i \leq m$ and $1 \leq j \leq n$) n = # mobile

$a_{i,j}$: approximate distance between mobile i and mobile j (for all $1 \leq i, j \leq n$)

Output: $x_1, \dots, x_n$: positions of mobiles

Objective Function: Minimize $\sum_{i,j} (\underbrace{d[p_i, x_j] - d_{i,j}}_{\alpha_{ij}})^2 + \sum_{i,j} (\underbrace{d[x_i, x_j] - a_{i,j}}_{\beta_{ij}})^2$ → not a linear function.

$$\boxed{\text{Minimize } \sum_{i,j} \alpha_{ij} + \sum_{i,j} \beta_{ij}} \rightarrow \text{linear}$$

Suppose that $h_{ij} = [d[p_i, x_j]]^2$ and $D_{ij} = \begin{bmatrix} 1 & p_{ij} \\ p_{ij} & h_{ij} \end{bmatrix}$

a variable to define later
free variable.

$$\det(D_{ij}) = h_{ij} - p_{ij}^2$$

In our SDP, we have $\boxed{D_{ij} \text{ be a semi-definite matrix.}}$

$$h_{ij} - p_{ij}^2 \geq 0$$
$$h_{ij} \geq p_{ij}^2$$
$$p_{ij} \leq \sqrt{h_{ij}}$$

By experiment, we usually have a matrix with smallest rank when the matrix is required to be SDP.

Minimum rank for D_{ij} : 1 $\Rightarrow \begin{bmatrix} p_{ij} \\ h_{ij} \end{bmatrix}$ should be multiple of $\begin{bmatrix} 1 \\ p_{ij} \end{bmatrix}$

$$\begin{bmatrix} p_{ij} \\ h_{ij} \end{bmatrix} = p_{ij} \begin{bmatrix} 1 \\ p_{ij} \end{bmatrix} = \begin{bmatrix} p_{ij} \\ p_{ij}^2 \end{bmatrix}$$

$h_{ij} = p_{ij}^2 \rightarrow \boxed{p_{ij} = \sqrt{h_{ij}}}$ $\rightarrow$ p_{ij} is free variable that will turn to be $\sqrt{h_{ij}} = d[p_i, x_j]$

$$\alpha_{ij} = (d[p_i, x_j] - d_{ij})^2 = d^2[p_i, x_j] - 2\, d_{ij} \cdot d[p_i, x_j] + d_{ij}^2$$

given number

$$\boxed{\alpha_{ij} = h_{ij} - 2 d_{ij}\, p_{ij} + d_{ij}^2}$$ linear constraint

---

How to denote $h_{ij} = d^2[p_i, x_j]$?

$$Z = \begin{bmatrix} 1 & 0 & \boxed{\text{x-axis of } x_1, \dots, x_n} \\ 0 & 1 & \boxed{\text{y-axis of } x_1 \dots x_n} \\ \text{x-axis of } x_1 \dots x_n & \text{y-axis} & ? \end{bmatrix}$$

$x_1' \dots x_n'$; $y_1' \dots y_n'$; $x_1' \dots x_n'$

free variables

In our SDP, we have $\boxed{Z \text{ be a semi-definite matrix}}$

Minimum rank for Z : 2

$$\begin{bmatrix} x_1' \\ y_1' \\ z_{11} \\ \vdots \\ z_{n1} \end{bmatrix} = x_1' \begin{bmatrix} 1 \\ 0 \\ x_1' \\ \vdots \\ x_n' \end{bmatrix} + y_1' \begin{bmatrix} 0 \\ 1 \\ y_1' \\ \vdots \\ y_n' \end{bmatrix}$$

$$Z = \begin{bmatrix} 1 & 0 & x_1' & \cdots & x_n' \\ 0 & 1 & y_1' & \cdots & y_n' \\ x_1' & y_1' & z_{11} & \cdots & z_{1n} \\ \vdots & \vdots & \vdots & & \\ x_n' & y_n' & z_{n1} & \cdots & z_{nn} \end{bmatrix}$$

bases of matrix

free variables

$$= \begin{bmatrix} x_1' \\ y_1' \\ x_1' \cdot x_1' + y_1' y_1' \\ x_1' \cdot x_2' + y_1' y_2' \\ \vdots \\ x_1' x_n' + y_1' y_n' \end{bmatrix}$$

$$z_{ij} = x_i' x_j' + y_i' y_j' \qquad [z_{ii} = x_i'^2 + y_i'^2$$

$$h_{ij} = (d[p_i, x_j])^2 = [x_i^p - x_j']^2 + [y_i^p - y_j']^2$$

$$= \underline{(x_i^p)^2} - \underline{2 x_i^p} x_j' + \boxed{x_j'^2} + \underline{(y_i^p)^2} - \underline{2 y_i^p} y_j' + \boxed{y_j^2} \rightarrow z_{jj}$$

given

$$\boxed{h_{ij} = z_{jj} - 2x_i^p x_j' - 2y_i^p y_j' + (x_i^p)^2 + (y_i^p)^2}$$

$$\beta_{ij} := (d[x_i, x_j] - a_{ij})^2$$

Suppose that $h_{ij}' = (d[x_i, x_j])^2$ and $U_{ij} = \begin{bmatrix} 1 & p_{ij}' \\ p_{ij}' & h_{ij}' \end{bmatrix}$.

We specify in SDP that U_{ij} is a semi-definite matrix.

$$p_{ij}' \approx \sqrt{h_{ij}'} = d[x_i, x_j]$$

$$\beta_{ij} = d^2[x_i, x_j] + \underline{2a_{ij}} d[x_i, x_j] + \underline{a_{ij}^2} = \boxed{h_{ij}' + 2a_{ij} p_{ij}' + a_{ij}^2 = \beta_{ij}}$$

given

$$h_{ij}' = (x_i' - x_j')^2 + (y_i' - y_j')^2 = x_i'^2 - 2x_i' x_j' + x_j'^2 + y_i'^2 - 2y_i' y_j' + y_j'^2$$

$$= \underbrace{x_i'^2 + y_i'^2}_{z_{ii}} \underbrace{- 2x_i' x_j' - 2y_i' y_j'}_{-2z_{i,j}} + \underbrace{y_j'^2 + y_j'^2}_{z_{j,j}}$$

$$\boxed{h_{ij}' = z_{ii} - 2z_{ij} + z_{jj}}$$

<u>Conclusion</u> minimize $\sum_{ij} \alpha_{ij} + \sum_{ij} \beta_{ij}$

such that $\begin{bmatrix} 1 & p_{ij} \\ p_{ij} & h_{ij} \end{bmatrix}$, $\begin{bmatrix} 1 & p_{ij}' \\ p_{ij}' & h_{ij}' \end{bmatrix}$ are semi-definite matrix.

$$\begin{bmatrix} 1 & 0 & x_1' & \cdots & x_n' \\ 0 & 1 & y_1' & \cdots & x_n' \\ x_1' & y_1' & z_{11} & \cdots & z_{1n} \\ \vdots & \vdots & \vdots & & \vdots \\ x_n' & y_n' & z_{n1} & \cdots & z_{nn} \end{bmatrix}$$ is a semi-definite matrix

$$\alpha_{ij} = h_{ij} - 2d_{ij} p_{ij} + d_{ij}^2$$

$$h_{ij} = z_{jj} - 2x_i^p x_j' - 2y_j^p y_j' + (x_i^p)^2 + (y_i^p)^2$$

$$\beta_{ij} = h_{ij}' + 2a_{ij} p_{ij}' + a_{ij}^2$$

$$h_{ij}' = z_{ii} - 2z_{ij} + z_{jj}$$