EE/ACM 150 - Applications of Convex Optimization in Signal Processing and Communications
Lecture 18

Andre Tkacenko

Signal Processing Research Group
Jet Propulsion Laboratory

May 31, 2012

Caltech

Outline

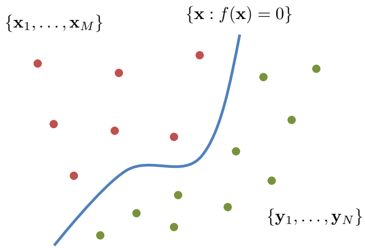
Introduction to Classification

In pattern recognition and classification problems, we are given two sets of points in $\mathbb{R}^n$, say $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ and $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$, and we wish to find a function $f : \mathbb{R}^n \to \mathbb{R}$ (say, within a given family of functions), such that we have the following.

$$f(\mathbf{x}_i) > 0\,,\ i = 1, \ldots, M\,,\ f(\mathbf{y}_i) < 0\,,\ i = 1, \ldots, N\,.$$

If these inequalities hold, we say that f (or specifically its 0-level set $\{\mathbf{x} : f(\mathbf{x}) = 0\}$) *separates*, *classifies*, or *discriminates* the two sets of points.

- If only weak versions of the inequalities hold (i.e., $\geq$ and $\leq$, respectively), then we refer to this as *weak separation*.
- This classification problem (including the weak relaxation) need not be feasible.



Linear Discrimination and Interpretations

In *linear discrimination*, we seek an affine function $f(\mathbf{x}) = \mathbf{a}^T\mathbf{x} - b$ that classifies the points, i.e.,

$$\mathbf{a}^T\mathbf{x}_i - b > 0\,,\ i = 1, \ldots, M\,,\ \mathbf{a}^T\mathbf{y}_i - b < 0\,,\ i = 1, \ldots, N\,.$$

Geometrically, we seek a hyperplane which separates the points. As the strict inequalities are *homogeneous* in $\mathbf{a}$ and b, they are feasible if and only if the following nonstrict inequalities

$$\mathbf{a}^T\mathbf{x}_i - b \geq 1\,,\ i = 1, \ldots, N\,,\ \mathbf{a}^T\mathbf{y}_i - b \leq -1\,,\ i = 1, \ldots, N\,,$$

are feasible in $\mathbf{a} \in \mathbb{R}^n$ and $b \in \mathbb{R}$. Note that such constraints are affine and, as such, convex.

Linear discrimination alternative interpretation:

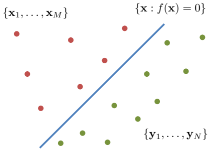
The strong alternative of the set of strict inequalities is the existence of $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ such that

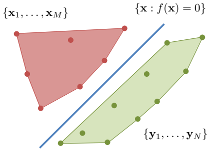
$$\boldsymbol{\lambda} \succeq \mathbf{0}\,,\ \boldsymbol{\mu} \succeq \mathbf{0}\,,\ (\boldsymbol{\lambda}, \boldsymbol{\mu}) \neq \mathbf{0}\,,\ \sum_{i=1}^{M} \lambda_i \mathbf{x}_i = \sum_{i=1}^{N} \mu_i \mathbf{y}_i\,,\ \mathbf{1}^T\boldsymbol{\lambda} = \mathbf{1}^T\boldsymbol{\mu}\,.$$

Here, $\boldsymbol{\lambda} \in \mathbb{R}^M$ and $\boldsymbol{\mu} \in \mathbb{R}^N$. Equivalently, this alternative can be expressed as follows.

$$\boldsymbol{\lambda} \succeq \mathbf{0}\,,\ \mathbf{1}^T\boldsymbol{\lambda} = 1\,,\ \boldsymbol{\mu} \succeq \mathbf{0}\,,\ \mathbf{1}^T\boldsymbol{\mu} = 1\,,\ \sum_{i=1}^{M} \lambda_i \mathbf{x}_i = \sum_{i=1}^{N} \mu_i \mathbf{y}_i\,.$$

This means there is a point in the convex hull of both $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ and $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$. So, two sets of points can be linearly discriminated if and only if their convex hulls do not intersect.



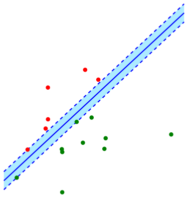


Robust Linear Discrimination

In *robust linear discrimination*, we seek to separate the two sets of points by the maximum possible margin hyperplane (or, equivalently, the thickest slab). As the Euclidean distance between the hyperplanes $\mathcal{H}_1 = \left\{ \mathbf{z} : \mathbf{a}^T \mathbf{z} - b = 1 \right\}$ and $\mathcal{H}_2 = \left\{ \mathbf{z} : \mathbf{a}^T \mathbf{z} - b = -1 \right\}$ is given by $\mathrm{dist}(\mathcal{H}_1, \mathcal{H}_2) = 2 / \left|\left|\mathbf{a}\right|\right|_2$, the maximum margin hyperplane can be found by solving the problem,

$$\begin{array}{ll} \text{minimize} & (1/2) \left|\left|\mathbf{a}\right|\right|_2 \\ \text{subject to} & \mathbf{a}^T \mathbf{x}_i - b \geq 1 \,,\ i = 1, \ldots, M \\ & \mathbf{a}^T \mathbf{y}_i - b \leq -1 \,,\ i = 1, \ldots, N \end{array} ,$$

which (after squaring the objective) becomes a QP in $\mathbf{a} \in \mathbb{R}^n$ and $b \in \mathbb{R}$.



Lagrange dual of maximum margin separation problem:

$$\begin{array}{ll} \text{maximize} & \mathbf{1}^T \boldsymbol{\lambda} + \mathbf{1}^T \boldsymbol{\mu} \\ \text{subject to} & 2 \left|\left| \sum_{i=1}^{M} \lambda_i \mathbf{x}_i - \sum_{i=1}^{N} \mu_i \mathbf{y}_i \right|\right|_2 \leq 1 \\ & \mathbf{1}^T \boldsymbol{\lambda} = \mathbf{1}^T \boldsymbol{\mu} \,,\ \boldsymbol{\lambda} \succeq \mathbf{0} \,,\ \boldsymbol{\mu} \succeq \mathbf{0} \end{array} .$$

From duality, the optimal value of the dual is the inverse of the maximum margin of separation.

Interpretation:

- Change variables to $\theta_i = \lambda_i / \mathbf{1}^T \boldsymbol{\lambda}$, $\gamma_i = \mu_i / \mathbf{1}^T \boldsymbol{\mu}$, and $t = 1 / \left(\mathbf{1}^T \boldsymbol{\lambda} + \mathbf{1}^T \boldsymbol{\mu} \right)$.
- Invert the objective to minimize $1 / \left(\mathbf{1}^T \boldsymbol{\lambda} + \mathbf{1}^T \boldsymbol{\mu} \right) = t$, yielding the following problem.

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \left|\left| \sum_{i=1}^{M} \theta_i \mathbf{x}_i - \sum_{i=1}^{N} \gamma_i \mathbf{y}_i \right|\right|_2 \leq t \\ & \boldsymbol{\theta} \succeq \mathbf{0} \,,\ \mathbf{1}^T \boldsymbol{\theta} = 1 \,,\ \boldsymbol{\gamma} \succeq \mathbf{0} \,,\ \mathbf{1}^T \boldsymbol{\gamma} = 1 \end{array} .$$

The optimal value of this problem is the distance between the convex hulls of the sets of points.

Approximate Linear Separation of Non-Separable Sets

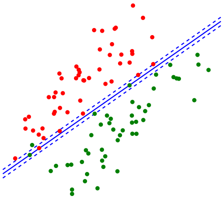
The two sets of points $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ and $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ need not be separable by a hyperplane, which occurs if their convex hulls intersect. In this case, we can relax the classification requirements $\mathbf{a}^T\mathbf{x}_i - b \geq 1$ for $i = 1, \ldots, M$ and $\mathbf{a}^T\mathbf{y}_i - b \leq -1$ for $i = 1, \ldots, N$ by introducing nonnegative variables $\mathbf{u} \in \mathbb{R}^M$ and $\mathbf{v} \in \mathbb{R}^N$ and forming the approximate separation conditions,

$$\mathbf{a}^T\mathbf{x}_i - b \geq 1 - u_i \,, \; i = 1, \ldots, M \,, \; \mathbf{a}^T\mathbf{y}_i - b \leq -\left(1 - v_i\right) \,, \; i = 1, \ldots, N \,.$$

These new conditions can always be made feasible by making $\mathbf{u}$ and $\mathbf{v}$ large enough. Our goal is to find $\mathbf{a}$, b, and *sparse* nonnegative $\mathbf{u}$ and $\mathbf{v}$ that satisfy these inequalities. As a heuristic for this, we can minimize the sum of the components of $\mathbf{u}$ and $\mathbf{v}$, which leads to the following problem.

$$\begin{array}{ll} \text{minimize} & \mathbf{1}^T\mathbf{u} + \mathbf{1}^T\mathbf{v} \\ \text{subject to} & \mathbf{a}^T\mathbf{x}_i - b \geq 1 - u_i \,, \; i = 1, \ldots, M \\ & \mathbf{a}^T\mathbf{y}_i - b \leq -\left(1 - v_i\right) \,, \; i = 1, \ldots, N \\ & \mathbf{u} \succeq \mathbf{0} \,, \; \mathbf{v} \succeq \mathbf{0} \end{array} .$$

- This problem is an LP in $\mathbf{a} \in \mathbb{R}^n$, $b \in \mathbb{R}$, $\mathbf{u} \in \mathbb{R}^M$, and $\mathbf{v} \in \mathbb{R}^N$.
- At an optimum, $u_i = \max\{0, 1 - \mathbf{a}^T\mathbf{x}_i + b\}$ and $v_i = \max\{0, 1 + \mathbf{a}^T\mathbf{y}_i - b\}$.
- It can be interpreted as a heuristic for minimizing the number of misclassified points.



Support Vector Machine

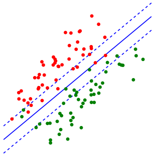
A *support vector machine* (SVM) (sometimes called a *support vector classifier*) represents a trade-off between the width of the slab $\left\{\mathbf{z} : -1 \leq \mathbf{a}^T\mathbf{z} - b \leq 1\right\}$ (given by $2/\left\|\mathbf{a}\right\|_2$) and the number of misclassified points. Specifically, an SVM combines robust linear discrimination with the approximate linear separation heuristic described above.

Support Vector Machine (SVM) Problem:

$$\begin{array}{ll} \text{minimize} & \left\|\mathbf{a}\right\|_2 + \gamma\left(\mathbf{1}^T\mathbf{u} + \mathbf{1}^T\mathbf{v}\right) \\ \text{subject to} & \mathbf{a}^T\mathbf{x}_i - b \geq 1 - u_i \, , \; i = 1, \ldots, M \\ & \mathbf{a}^T\mathbf{y}_i - b \leq -\left(1 - v_i\right) \, , \; i = 1, \ldots, N \\ & \mathbf{u} \succeq \mathbf{0} \, , \; \mathbf{v} \succeq \mathbf{0} \end{array} .$$

- Here, $\gamma > 0$ is the trade-off parameter between slab width and # of misclassified points.
- The solution produces a point on the optimal trade-off curve between inverse of margin $2/\left\|\mathbf{a}\right\|_2$ and classification error, measured by the total slack $\mathbf{1}^T\mathbf{u} + \mathbf{1}^T\mathbf{v}$.

Same example data as in previous slide, with $\gamma = 0.1$:



Linear Discrimination and Logistic Modeling

Another approach to approximately classify the two sets of points when they cannot be linearly separated is based on the *logistic model*. Specifically, we fit the sets of points to a logistic model. Let $\mathrm{z} \in \{0, 1\}$ be a random variable with

$$\begin{aligned}\Pr\{\mathrm{z} = 0\} &= 1/\left(1 + \exp\left(\mathbf{a}^T\mathbf{u} - b\right)\right) , \\ \Pr\{\mathrm{z} = 1\} &= \left(\exp\left(\mathbf{a}^T\mathbf{u} - b\right)\right) / \left(1 + \exp\left(\mathbf{a}^T\mathbf{u} - b\right)\right) ,\end{aligned}$$

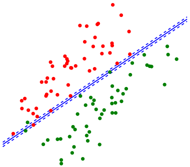
where $\mathbf{u} \in \mathbb{R}^n$ is the explanatory variable of the logistic model. Then, $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ and $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ are the respective values of $\mathbf{u}$ for the M and N samples for which $\mathrm{z} = 0$ and $\mathrm{z} = 1$. We can then determine $\mathbf{a}$ and b using ML estimation, by solving the convex optimization problem

$$\text{minimize} \quad -\ell(\mathbf{a}, b) \quad ,$$

with variables $\mathbf{a} \in \mathbb{R}^n$ and $b \in \mathbb{R}$, where ℓ is the log-likelihood function

$$\ell(\mathbf{a}, b) = -\sum_{i=1}^{M} \log\left(1 + \exp\left(\mathbf{a}^T\mathbf{x}_i - b\right)\right) + \sum_{i=1}^{N}\left(\mathbf{a}^T\mathbf{y}_i - b\right) - \sum_{i=1}^{N} \log\left(1 + \exp\left(\mathbf{a}^T\mathbf{y}_i - b\right)\right) .$$

- If the data points truly come from a logistic model, then the affine classifier $f(\mathbf{z}) = (\mathbf{a}^\star)^T\mathbf{z} - b^\star$ has the smallest probability of misclassification among all affine classifiers.
- The hyperplane $\mathbf{a}^T\mathbf{u} = b$ corresponds to the points where $\Pr\{\mathrm{z} = 0\} = \Pr\{\mathrm{z} = 1\} = 1/2$, i.e, the two outcomes are equally likely.



Introduction to Nonlinear Discrimination

In many cases, we can separate two sets of points $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ and $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ by a *nonlinear* function $f : \mathbb{R}^n \to \mathbb{R}$ as follows.

$$f(\mathbf{x}_i) > 0\,,\ i = 1, \ldots, M\,,\ f(\mathbf{y}_i) < 0\,,\ i = 1, \ldots, N\,.$$

Linearly parameterized family of functions:

- One way to do this is to choose a linearly parametrized family of functions

  $$f(\mathbf{z}) = \boldsymbol{\theta}^T \mathbf{f}(\mathbf{z})\ ,$$

  where $\boldsymbol{\theta} \in \mathbb{R}^k$ is a parameter vector and $\mathbf{f}(\mathbf{z}) \triangleq (f_1(\mathbf{z})\,, \ldots, f_k(\mathbf{z})) : \mathbb{R}^n \to \mathbb{R}^k$ is a vector of *basis functions*.
- With this linearly parametrized construction, the above inequalities can be solved in the exactly the same way as in linear discrimination.

  Specifically, as f is homogeneous in $\boldsymbol{\theta}$, we solve a set of affine inequalities in $\boldsymbol{\theta}$:

  $$\boldsymbol{\theta}^T \mathbf{f}(\mathbf{x}_i) \geq 1\,,\ i = 1, \ldots, M\,,\ \boldsymbol{\theta}^T \mathbf{f}(\mathbf{y}_i) \leq -1\,,\ i = 1, \ldots, N\,.$$

Quadratic Discrimination

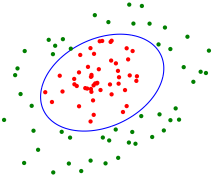
Suppose that $f(\mathbf{z}) = \mathbf{z}^T \mathbf{P} \mathbf{z} + \mathbf{q}^T \mathbf{z} + r$, where the parameters to be chosen are $\mathbf{P} \in \mathbb{S}^n$, $\mathbf{q} \in \mathbb{R}^n$, and $r \in \mathbb{R}$. Here, we must satisfy the affine inequalities

$$\mathbf{x}_i^T \mathbf{P} \mathbf{x}_i + \mathbf{q}^T \mathbf{x}_i + r \geq 1 \,,\ i = 1, \ldots, M \,,\ \mathbf{y}_i^T \mathbf{P} \mathbf{y}_i + \mathbf{q}^T \mathbf{y}_i + r \leq -1 \,,\ i = 1, \ldots, N \,.$$

We can also pose additional constraints. For example, we can require $\mathbf{P} \prec \mathbf{0}$, which means that the separating surface is ellipsoidal. In other words, we seek an ellipsoid containing all the points $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$, but none of the points $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$. This quadratic discrimination problem can then be solved as an SDP feasibility problem:

$$\begin{array}{ll} \text{find} & \mathbf{P}, \mathbf{q}, r \\ \text{subject to} & \mathbf{x}_i^T \mathbf{P} \mathbf{x}_i + \mathbf{q}^T \mathbf{x}_i + r \geq 1 \,,\ i = 1, \ldots, M \\ & \mathbf{y}_i^T \mathbf{P} \mathbf{y}_i + \mathbf{q}^T \mathbf{y}_i + r \leq -1 \,,\ i = 1, \ldots, N \\ & \mathbf{P} \preceq -\mathbf{I} \end{array} ,$$

with variables $\mathbf{P} \in \mathbb{S}^n$, $\mathbf{q} \in \mathbb{R}^n$, and $r \in \mathbb{R}$. (Here, we use homogeneity in $\mathbf{P}$, $\mathbf{q}$, and r to express the constraint $\mathbf{P} \prec \mathbf{0}$ as $\mathbf{P} \preceq -\mathbf{I}$.)



Polynomial Discrimination

Consider the set of polynomials on $\mathbb{R}^n$ with degree less than or equal to d:

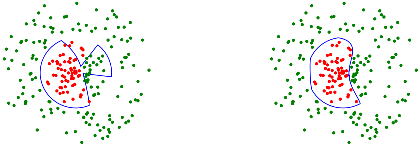
$$f(\mathbf{z}) = \sum_{i_1+\cdots+i_n \le d} a_{i_1,\ldots,i_d} z_1^{i_1} \cdots z_n^{i_n} \,.$$

Note that this can be expressed as $f(\mathbf{z}) = \boldsymbol{\theta}^T \mathbf{f}(\mathbf{z})$, where $\boldsymbol{\theta}$ is a vector of all the coefficients $a_{i_1,\ldots,i_d}$ and $\mathbf{f}(\mathbf{z})$ consists of all monomials up to the given degree of d.

- By solving the discrimination feasibility problem

$$\begin{array}{ll} \text{find} & \boldsymbol{\theta} \\ \text{subject to} & \boldsymbol{\theta}^T \mathbf{f}(\mathbf{x}_i) \ge 1 \,,\ i = 1, \ldots, M \\ & \boldsymbol{\theta}^T \mathbf{f}(\mathbf{y}_i) \le -1 \,,\ i = 1, \ldots, N \end{array} \,,$$

  we check, geometrically, whether the two sets can be separated by an algebraic surface.
- As the degree is a quasiconvex function of the coefficients, we can find the minimum degree polynomial on $\mathbb{R}^n$ that separates two sets of points via quasiconvex programming.



No cubic polynomial can separate the points (left), whereas a quartic polynomial can do so (right).

Introduction to Placement and Location Problems

- We are given N points with coordinates $\mathbf{x}_i \in \mathbb{R}^k$, where $k = 2$ or $k = 3$.
- Some of the locations $\mathbf{x}_i$ are given, while the other positions $\mathbf{x}_i$ are variables.
- For each pair of points, we associate a cost function $f_{i,j}(\mathbf{x}_i, \mathbf{x}_j)$.

Placement Problem:

$$\text{minimize} \quad \sum_{(i,j)\in\mathcal{A}} f_{i,j}(\mathbf{x}_i, \mathbf{x}_j) \quad .$$

- Here, $\mathcal{A}$ is the set of all links in the graph described by the N points or nodes.
- Alternatively, the objective can be expressed as a sum over all arcs (i, j) if we set $f_{i,j} = 0$ when links i and j are not connected.
- The variables are the positions of the free points.

Interpretations:

- The points may represent plants or warehouses; $f_{i,j}$ could be the transportation cost between facilities i and j.
- The points may represent cells on an integrated circuit (IC); $f_{i,j}$ could then represent the wirelength.

Linear Facility Location Problems

The simplest version of the placement problem occurs when the cost function $f_{i,j}$ is a linear function of the distance between nodes i and j. In other words, we have

$$f_{i,j}(\mathbf{x}_i, \mathbf{x}_j) = w_{i,j} \left|\left|\mathbf{x}_i - \mathbf{x}_j\right|\right| \ ,$$

where $w_{i,j} \geq 0$ is a weight parameter. By proper choice of the weights $w_{i,j}$, the placement problem objective can be expressed in the following two ways.

$$\sum_{(i,j)\in\mathcal{A}} w_{i,j} \left|\left|\mathbf{x}_i - \mathbf{x}_j\right|\right| = \sum_{i<j} w_{i,j} \left|\left|\mathbf{x}_i - \mathbf{x}_j\right|\right| \ .$$

For this choice of cost function, the placement problem is always convex.

Examples: *One free point* $(u, v) \in \mathbb{R}^2$ *with fixed points* $(u_1, v_1), \ldots, (u_K, v_K)$*.*

- ℓ_1*-norm:* When $w_{i,j} = 1$ for all i, j, the objective is

  $$\sum_{i=1}^{K} \left(|u - u_i| + |v - v_i|\right) \ .$$

  An optimal point is any *median* of the fixed points. That is, u and v can be taken to be any median of the points $\{u_1, \ldots, u_K\}$ and $\{v_1, \ldots, v_K\}$, respectively.
- *Euclidean norm:* When $w_{i,j} = 1$ for all i, j, the objective is

  $$\sum_{i=1}^{K} \left((u - u_i)^2 + (v - v_i)^2\right)^{1/2} \ .$$

  The point (u, v) that minimizes this is called the *geometric median* or the *Fermat-Weber point* of the given fixed points.

Types of Placement Constraints

There are several interesting constraints that can be included in the placement problem which preserve convexity.

- We can require some positions $\mathbf{x}_i$ to lie in a specified convex set, e.g., a particular line, interval, square, or ellipsoid.
- The relative position of one point with respect to one or more other points can be constrained. For example, we can limit the distance between a pair of points.
- We can impose relative position constraints, e.g., that one point must lie to the left of another point.
- Several *bounding box* constraints can be included. The bounding box of a group of points is the smallest rectangle that contains the points. For example, we can constrain the points $\mathbf{x}_1, \ldots, \mathbf{x}_p$ to lie in a bounding box with perimeter not exceeding $P_{\max}$ by adding the constraints

$$\mathbf{u} \preceq \mathbf{x}_i \preceq \mathbf{v}\,,\ i = 1, \ldots, p\,,\ 2\mathbf{1}^T(\mathbf{v} - \mathbf{u}) \leq P_{\max}\,,$$

where $\mathbf{u} \in \mathbb{R}^k$ and $\mathbf{v} \in \mathbb{R}^k$ are additional variables.

Nonlinear Facility Location Problems

A generalization of the linear facility location problem occurs when we take the cost function to be

$$f_{i,j}(\mathbf{x}_i, \mathbf{x}_j) = w_{i,j} h(||\mathbf{x}_i - \mathbf{x}_j||) \ ,$$

where $w_{i,j} \geq 0$ is a weight parameter and h is a nondecreasing (on $\mathbb{R}_+$) and convex function. In this case, we call the associated problem a *nonlinear placement* or *facility location problem*.

Examples:

- *Quadratic placement problem:* When the Euclidean norm is used with $h(z) = z^2$, we obtain the *quadratic placement problem*. As an example, when there is one free point $\mathbf{x}$ and fixed points $\mathbf{x}_1, \ldots, \mathbf{x}_K$, and $w_{i,j} = 1$ for all i, j, the placement problem becomes

  $$\text{minimize} \quad ||\mathbf{x} - \mathbf{x}_1||_2^2 + \cdots + ||\mathbf{x} - \mathbf{x}_K||_2^2 \quad .$$

  This has the optimal solution $\mathbf{x}^\star = (\mathbf{x}_1 + \cdots + \mathbf{x}_K)/K$, which is the average or centroid of the fixed points.
- *Power function:* (with power $p \geq 1$)

  $$h(z) = z^p \, .$$

- *Deadzone-linear function:* (with deadzone width 2γ)

  $$h(z) = \begin{cases} 0 \, , & |z| \leq \gamma \\ |z - \gamma| \ , & |z| > \gamma \end{cases} \quad .$$

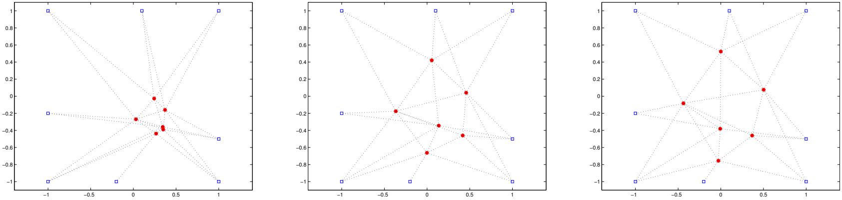
- *Quadratic-linear function:* (with quadratic width 2γ)

  $$h(z) = \begin{cases} z^2 \, , & |z| \leq \gamma \\ 2\gamma |z| - \gamma^2 \, , & |z| > \gamma \end{cases} \quad .$$

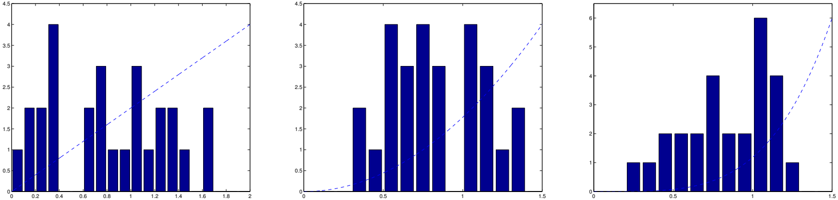
Comparison Between Linear and Nonlinear Placement

Example: Minimize $\sum_{(i,j)\in\mathcal{A}} h\big(||\mathbf{x}_i - \mathbf{x}_j||_2\big)$, with 6 free points (red dots), 8 fixed points (blue squares), and 27 links (black dotted lines).

Optimal placement for $h(z) = z$, $h(z) = z^2$, and $h(z) = z^4$ (from left to right, respectively).



Histograms of respective connection lengths $||\mathbf{x}_i - \mathbf{x}_j||_2$ for $h(z) = z$, $h(z) = z^2$, and $h(z) = z^4$.



Path Constraints

A p-link *path* along the points $\mathbf{x}_1, \ldots, \mathbf{x}_N$ is described by a sequence of nodes, $i_0, \ldots, i_p \in \{1, \ldots, N\}$. The length of the path is given by

$$\left|\left|\mathbf{x}_{i_1} - \mathbf{x}_{i_0}\right|\right| + \left|\left|\mathbf{x}_{i_2} - \mathbf{x}_{i_1}\right|\right| + \cdots + \left|\left|\mathbf{x}_{i_p} - \mathbf{x}_{i_{p-1}}\right|\right| .$$

- The path length is a convex function of $\mathbf{x}_1, \ldots, \mathbf{x}_N$.
- As a result of the convexity of the path length, imposing an upper bound on the length of a path is a convex constraint.
- Several interesting placement problems involve path constraints, or have an objective based on path lengths.
- One typical example is the *minimax delay placement problem*, in which the objective is based on a maximum path length over a set of paths.

Minimax Delay Placement Problem

To introduce the minimax delay placement problem, we first describe the topology characterizing the problem.

- We consider a directed, acyclic graph with nodes $1, \ldots, N$, and arcs or links represented by a set $\mathcal{A}$ of ordered pairs: $(i, j) \in \mathcal{A}$ if and only if an arc points from i to j.
- Node i is a *source node* if no arc $\mathcal{A}$ points to it; it is a *sink node* or *destination node* if no arc in $\mathcal{A}$ leaves from it.
- The arcs model a kind of flow, say of goods or information, in a network with nodes at positions $\mathbf{x}_1, \ldots, \mathbf{x}_N$. The flow starts at a source node, moves along a path from node to node, and ends up at a sink or destination node.
- We use the distance between successive nodes to model propagation time, or shipment time, of goods between nodes; the total delay or propagation time of a path is then proportional to the sum of the distances between successive nodes.
- Some node locations are fixed, while others are free. The goal is to choose the free node locations to minimize the maximum total delay, for any path from a source node to a sink node. This leads to a convex problem, since the objective

$$T_{\max} \triangleq \max\left\{ ||\mathbf{x}_{i_1} - \mathbf{x}_{i_0}|| + \cdots + \left|\left|\mathbf{x}_{i_p} - \mathbf{x}_{i_{p-1}}\right|\right| : i_0, \ldots, i_p \text{ is a source-sink path} \right\} ,$$

  is a convex function of the locations $\mathbf{x}_1, \ldots, \mathbf{x}_N$.

Minimax Delay Placement Problem (Continued)

- While the minimax delay placement problem is convex, the number of source-sink paths is exponential in the number of nodes or arcs, which can become quite large.
- However, the problem can be reformulated to avoid enumerating all sink-source paths.
- To show this, let τ_k denote the maximum total delay of any path from node k to a sink node. Suppose node k has outgoing arcs to nodes $j_1, \ldots, j_p$. Then, based on a simple *dynamic programming* argument, we can show that

$$\tau_k = \max\left\{||\mathbf{x}_{j_1} - \mathbf{x}_k|| + \tau_{j_1}, \ldots, \left|\left|\mathbf{x}_{j_p} - \mathbf{x}_k\right|\right| + \tau_{j_p}\right\} .$$

- The above equation gives a recursion for finding the maximum delay from any node. Specifically, we start at the sink nodes, and work backward until we reach all source nodes.
- The number of arithmetic operations required for this is approximately the number of links.
- With this recursion, the minimax delay placement problem can be expressed as

$$\begin{array}{ll} \text{minimize} & \max\{\tau_k : k \text{ is a source node}\} \\ \text{subject to} & \tau_k = 0\,,\ k \text{ is a sink node} \\ & \tau_k = \max\{||\mathbf{x}_j - \mathbf{x}_k|| + \tau_j : \text{there is an arc from } k \text{ to } j\} \end{array} .$$

  This form of the problem, however, is not convex.
- By replacing the equality constraints with inequalities, we obtain an equivalent form of the problem that is convex. Introducing new variables $T_1, \ldots, T_N$ as upper bounds on $\tau_1, \ldots, \tau_N$, the minimax delay placement problem can be expressed as the convex problem,

$$\begin{array}{ll} \text{minimize} & \max\{T_k : k \text{ is a source node}\} \\ \text{subject to} & T_k = 0\,,\ k \text{ is a sink node} \\ & T_k \geq \max\{||\mathbf{x}_j - \mathbf{x}_k|| + T_j : \text{there is an arc from } k \text{ to } j\} \end{array} .$$

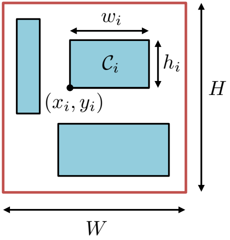
Introduction to Floor Planning

A *floor planning problem* is an extension of a placement problem in the following ways:

- The objects to be placed are rectangles or boxes aligned with the axes (as opposed to points), and must not overlap.
- Each rectangle or box to be placed can be reconfigured, within some limits.

Typically, the objective is to minimize the size (e.g., area, volume, or perimeter) of the *bounding box*, i.e., the smallest box containing the boxes to be configured and placed.

- The non-overlap constraints make the general floor planning problem a complicated combinatorial optimization problem or rectangular packing problem.
- However, if the *relative positioning* of the boxes is specified, then several types of floor planning problems can be formulated as convex optimization problems.



Introduction to Floor Planning (Continued)

We will focus on the two-dimensional case of floor planning here.

Assumptions and required constraints:

- We have N cells or modules $\mathcal{C}_1, \ldots, \mathcal{C}_N$ that are to be configured and placed in a rectangle with width W and height H, and lower left corner at the origin $(0, 0)$.
- The geometry and position of the i-th cell is specified by its width w_i and height h_i, and the coordinates (x_i, y_i) of its lower left corner.
- The variables of the problem are x_i, y_i, h_i, w_i for $i = 1, \ldots, N$, and the width W and height H of the bounding rectangle.
- We require the cells to lie inside the bounding rectangle, i.e.,

$$x_i \geq 0\,,\ y_i \geq 0\,,\ x_i + w_i \leq W\,,\ y_i + h_i \leq H\,,\ i = 1, \ldots, N\,.$$

- We also require that the cells do not overlap, expect possibly on their boundaries:

$$\mathrm{int}(\mathcal{C}_i \cap \mathcal{C}_j) = \varnothing \text{ for } i \neq j\,.$$

(A minimum clearance can be included as well.) This holds if and only if, for $i \neq j$,

$\mathcal{C}_i$ is left of $\mathcal{C}_j$, or $\mathcal{C}_i$ is right of $\mathcal{C}_j$, or $\mathcal{C}_i$ is below $\mathcal{C}_j$, or $\mathcal{C}_i$ is above $\mathcal{C}_j$.

These geometric conditions correspond to the following inequalities, for $i \neq j$.

$$x_i + w_i \leq x_j\,, \text{ or } x_j + w_j \leq x_i\,, \text{ or } y_i + h_i \leq y_j\,, \text{ or } y_j + h_j \leq y_i\,.$$

Relative Positioning Constraints

To specify relative positioning constraints, we define relations $\mathcal{L}$ and $\mathcal{B}$ on pairs of nodes which specify whether cells are to the left of or below each other, respectively.

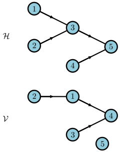
- We have that $\mathcal{C}_i$ is to the left of $\mathcal{C}_j$ if $(i, j) \in \mathcal{L}$, and similarly $\mathcal{C}_i$ is below $\mathcal{C}_j$ if $(i, j) \in \mathcal{B}$.
- This yields the following constraints for $i, j = 1, \ldots, N$.

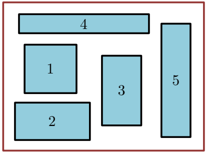
  $$x_i + w_i \leq x_j \text{ for } (i, j) \in \mathcal{L}\,,\ y_i + h_i \leq y_j \text{ for } (i, j) \in \mathcal{B}\,.$$

  We require that for each (i, j) with $i \neq j$, one of the following must hold:

  $$(i, j) \in \mathcal{L}\,,\ (j, i) \in \mathcal{L}\,,\ (i, j) \in \mathcal{B}\,,\ (j, i) \in \mathcal{B}\,.$$

  In addition, $(i, i) \notin \mathcal{L}$ and $(i, i) \notin \mathcal{B}$. This leads to a set of $N(N-1)/2$ affine inequalities.
- The relations $\mathcal{L}$ and $\mathcal{B}$ are anti-symmetric (i.e., $(i, j) \in \mathcal{L} \Rightarrow (j, i) \notin \mathcal{L}$) and transitive (i.e., $(i, j) \in \mathcal{L}\,,\ (j, k) \in \mathcal{L} \Rightarrow (i, k) \in \mathcal{L}$). Transitivity leads to removing redundant constraints.
- A minimal set of relative positioning constraints can be described using two directed acyclic graphs $\mathcal{H}$ (for horizontal) and $\mathcal{V}$ (for vertical). We have $(i, j) \in \mathcal{L}$ (similarly $(i, j) \in \mathcal{B}$) if and only if there is a directed path in $\mathcal{H}$ (similarly $\mathcal{V}$) from i to j.
- To ensure a relative positioning constraint for every pair of cells, we require that for every pair, there must be a directed path from one cell to the other in one of the graphs.





Relative Positioning Constraints (Continued)

- We only need to impose the inequalities corresponding to the edges of the graphs $\mathcal{H}$ and $\mathcal{V}$; the others follow from transitivity.
- So, for relative positioning, we need only enforce the following set of inequalities.

$$x_i + w_i \le x_j \text{ for } (i,j) \in \mathcal{H}\,,\ y_i + h_i \le y_j \text{ for } (i,j) \in \mathcal{V}\,.$$

- In a similar way, the $4N$ bounding box inequalities,

$$x_i \ge 0\,,\ y_i \ge 0\,,\ x_i + w_i \le W\,,\ y_i + h_i \le H\,,\ i = 1, \ldots, N\,,$$

  can be reduced to a minimal equivalent set.
- For example, the constraint $x_i \ge 0$ need only be imposed on the left-most cells. Similarly, the inequalities $x_i + w_i \ge W$ need only be enforced for the right-most cells. Analogous results hold for the vertical bounding box inequalities.
- This leads to the following minimal equivalent set of bounding box inequalities.

$$\begin{aligned} x_i \ge 0 \text{ for } i\ \mathcal{L} \text{ minimal}, &\quad x_i + w_i \le W \text{ for } i\ \mathcal{L} \text{ maximal}, \\ y_i \ge 0 \text{ for } i\ \mathcal{B} \text{ minimal}, &\quad y_i + h_i \le H \text{ for } i\ \mathcal{B} \text{ maximal}. \end{aligned}$$

For the example above, a minimal set of horizontal relative positioning inequalities is given by

$$\begin{gathered} x_1 \ge 0\,,\ x_2 \ge 0\,,\ x_4 \ge 0\,,\ x_5 + w_5 \le W\,,\ x_1 + w_1 \le x_3\,, \\ x_2 + w_2 \le x_3\,,\ x_3 + w_3 \le x_5\,,\ x_4 + w_4 \le x_5\,. \end{gathered}$$

Similarly, a minimal set of vertical relative positioning inequalities is given by

$$\begin{gathered} y_2 \ge 0\,,\ y_3 \ge 0\,,\ y_5 \ge 0\,,\ y_4 + h_4 \le H\,,\ y_5 + h_5 \le H\,, \\ y_2 + h_2 \le y_1\,,\ y_1 + h_1 \le y_4\,,\ y_3 + h_3 \le y_4\,. \end{gathered}$$

Floor Planning Convex Constraints

For all floor planning problems considered here, we will take the objective to be the perimeter of the bounding box, i.e., $2(W+H)$, which is a linear function of the variables. A list of several convex constraints that can be added to the floor planning problem in addition to the convex relative positioning and bounding box constraints is as follows.

- *Minimum spacing:* ($\rho^{\mathrm{h}}_{i,j} \geq 0, \rho^{\mathrm{v}}_{i,j} \geq 0$)

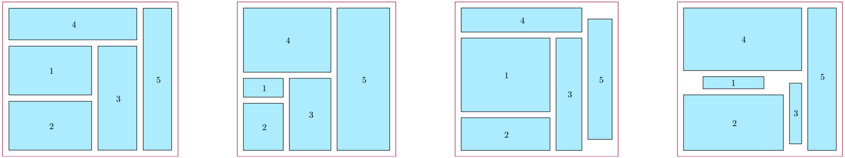
$$x_i + w_i + \rho^{\mathrm{h}}_{i,j} \leq x_j \text{ for } (i,j) \in \mathcal{H}\ , \text{ and/or } y_i + h_i + \rho^{\mathrm{v}}_{i,j} \leq y_j \text{ for } (i,j) \in \mathcal{V}\,.$$

- *Minimum cell area:* ($A_i \geq 0$)

$$w_i h_i \geq A_i \iff w_i \geq A_i/h_i \iff (w_i h_i)^{1/2} \geq A_i^{1/2} \iff \log w_i + \log h_i \geq \log A_i\,.$$

- *Aspect ratio constraints:* ($u_i \geq l_i \geq 0$)

$$l_i \leq w_i/h_i \leq u_i\,.$$



Examples using above ordering constraints, with the same minimum required spacing of $\rho = 1$, and the same aspect ratio constraint $1/5 \leq w_i/h_i \leq 5$, but different minimum required cell areas A_i. Here, the values of A_i are chosen such that $\sum_{i=1}^{5} A_i$ is the same for each case.

Floor Planning Convex Constraints (Continued)

- *Alignment constraints:* We can impose that a cell is aligned with another cell or the boundary box in a certain way. All such conditions are affine equality constraints.
- *Symmetry constraints:* We can require pairs of cells to be symmetric about a vertical or horizontal axis, that can be fixed or floating. This leads to affine equality constraints.
- *Similarity constraints:* We can impose the width and/or height of cell i to be a scale factor, say a, of cell j, by imposing the linear equality constraints $w_i = aw_j$ and/or $h_i = ah_j$.
- *Containment constraints:* We can require that a particular cell contains a given point or lies inside a given polyhedron, by imposing affine inequality constraints.
- *Distance constraints:* Several convex distance constraints can be imposed. For example, to limit the distance between the centers of cells i and j, we use the convex inequality

$$||(x_i + w_i/2, y_i + h_i/2) - (x_j + w_j/2, y_j + h_j/2)|| \leq D_{i,j} \, .$$

  We can also require $\mathrm{dist}(\mathcal{C}_i, \mathcal{C}_j) \leq D_{i,j}$, by introducing four new variables u_i, v_i, u_j, v_j, and imposing the affine inequalities

$$x_i \leq u_i \leq x_i + w_i \, , \; y_i \leq v_i \leq y_i + h_i \, ,$$

  along with the convex inequality

$$||(u_i, v_i) - (u_j, v_j)|| \leq D_{i,j} \, .$$

  - *ℓ_∞-norm:* We have $\mathrm{dist}(\mathcal{C}_i, \mathcal{C}_j) \leq D_{i,j}$ if and only if $x_j - (x_i + w_i) \leq D_{i,j}$, $y_j - (y_i + h_i) \leq D_{i,j}$, and $y_i - (y_j + h_j) \leq D_{i,j}$.
  - *ℓ_1-norm or ℓ_2-norm:* We have $\mathrm{dist}(\mathcal{C}_i, \mathcal{C}_j) \leq D_{i,j}$ if and only if $y_j - (y_i + h_i) \leq d_v$, $y_i - (y_j + h_j) \leq d_v$, $d_v \geq 0$, and $x_j - (x_i + w_i) + d_v \leq D_{i,j}$ for the ℓ_1-norm or $(x_j - (x_i + w_i))^2 + d_v^2 \leq D_{i,j}^2$ for the ℓ_2-norm.

Floor Planning Via Geometric Programming

The floor planning problem can also be formulated as a GP in the variables x_i, y_i, w_i, h_i, W, H.

Relative positioning and bounding box constraints:

- The relative positioning constraints can be expressed in the posynomial form,
$$x_i x_j^{-1} + w_i x_j^{-1} \leq 1 \text{ for } (i,j) \in \mathcal{H}\,,\ y_i y_j^{-1} + h_i y_j^{-1} \leq 1 \text{ for } (i,j) \in \mathcal{V}\,.$$
- The bounding box constraints have the following posynomial form.
$$x_i W^{-1} + w_i W^{-1} \leq 1 \text{ for } i\ \mathcal{L} \text{ maximal}\,,\ y_i H^{-1} + h_i H^{-1} \leq 1 \text{ for } i\ \mathcal{B} \text{ maximal}\,.$$
Note that here, the constraints that $x_i \geq 0$ for i $\mathcal{L}$ minimal and $y_i \geq 0$ for i $\mathcal{B}$ minimal are implicit as a result of the domain of the GP problem formulation.

Nuances of GP formulation of floor planning problem:

- We can minimize the bounding box area WH, since it is a posynomial.
- We can exactly specify the area of any cell, since $w_i h_i = A_i$ is a monomial equality constraint.
- Alignment, symmetry, and distance constraints cannot be handled with the GP formulation.
- Similarity can be handled with the GP formulation, where the scaling ratio can either be given or unspecified (in which case it can be treated as another variable).