Planar Maximal Covering with Ellipses

Danilo F. Tedeschi

Instituto de Ciências Matemáticas e Computação

April 25, 2019

Contents

Introduction

- Covering problems
  - Set Cover Problem
  - Maximal Covering Problem
- Maximal Covering Location Problem (MCLP)
- Planar Maximal Covering Location Problem (PMCLP)
  - One disk: $\mathcal{O}(n^2)$ and $\mathcal{O}(n^2 \log n)$ algorithms
  - m disks: $\mathcal{O}(n^{2m-1} \log n)$ algorithm
- Goals
  - Develop a $\mathcal{O}(n^2 \log n)$ algorithm for the one disk case
  - Adapt it for the m ellipses case

Preliminaries

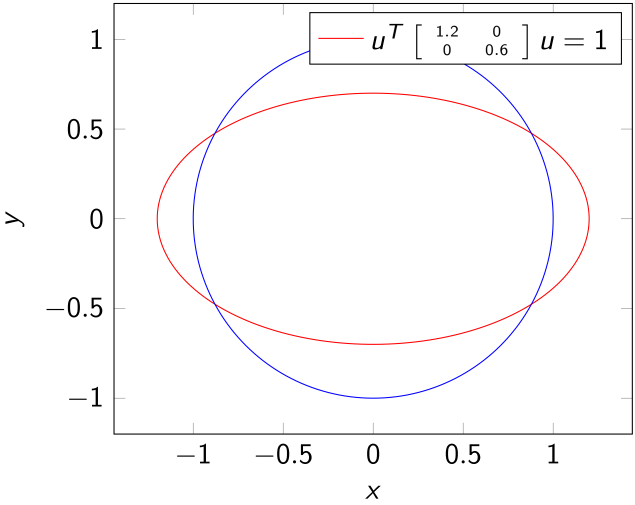
Norms

Let $u \in \mathbb{R}^2$ and Q a 2x2 positive definite matrix

- Euclidean
$$||u||_2 = \sqrt{u^T u}$$
- Elliptical
$$||u||_Q = \sqrt{u^T Q u}$$

Preliminaries

Figure: The elliptical and euclidean norms.



Preliminaries

Ellipse

Given a center $c \in \mathbb{R}^2$ and a 2x2 p.d. matrix Q, an ellipse is the set of points that satisfy

$$||u - c||_Q = 1,$$

with $\leq$ representing the set of covered points

Axis-parallel ellipse

Any 2 by 2 diagonal d.p. matrix determines an axis-parallel ellipse, which can also be described by

$$\frac{(x - c_x)^2}{a^2} + \frac{(y - c_y)^2}{b^2} = 1,$$

where (a, b) are the shape parameters and $c = (c_x, c_y)$ is the center.

$MCD(\mathscr{P}, 1)$ is the problem of placing one disk on the plane to cover a subset of a demand set $\mathscr{P}$ maximizing the weights of the covered points.

$$\max_q w(\mathscr{P} \cap D(q)),$$

- $\mathscr{P} = \{p_1, \ldots, p_n\}$ is the demand set with weights $w(p_i) > 0$
- $w(A)$, $A \subset \mathscr{P}$, is the sum of weights of the points in A
- $D(q)$ is a unit disk with center at point q
- We will introduce an equivalent problem...

Maximum Weight Clique Problem

Let $\mathscr{D} = \{D_1, \ldots, D_n\}$ be a set of n unit disks with weights $w_i > 0$. The maximum weight clique is defined as

$$\max_{q} \sum_{D_k \cap q \neq \emptyset} w_k,$$

- The disks are fixed with centers at $\mathscr{P} = \{p_1, \ldots, p_n\}$ with $w_k = w(p_k)$
- A clique is a non-empty intersection area of a subset of disks. We search for only a point in an optimal clique.
- An optimal solution for the maximum weight clique is an optimal solution for $MCD(\mathscr{P}, 1)$.

Figure: An instance of $MCD(\mathscr{P}, 1)$.

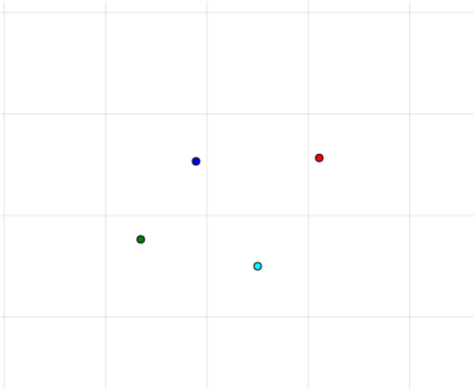
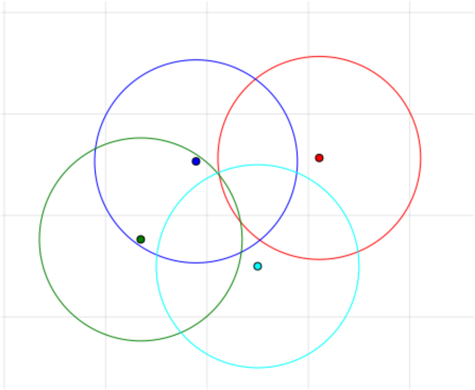
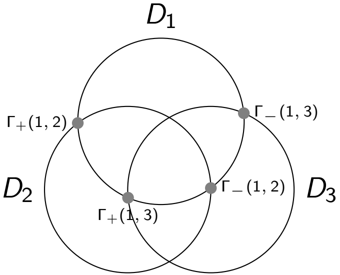


Figure: An instance of $MCD(\mathscr{P}, 1)$.



Let $\Gamma_+(i,j)$ and $\Gamma_-(i,j)$ be the opening and closing angles of intersections of disks i and j. Also, $\Gamma_+(i,j), \Gamma_-(i,j) \in [0, 2\pi]$.

Figure: Three disks and their intersection points.

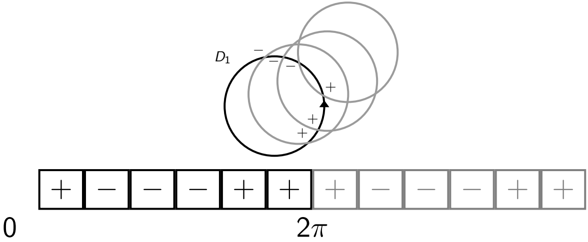


For a disk D_i, a counter-clockwise traversal visits every $\Gamma_+(i,j)$ and $\Gamma_-(i,j)$ in counter-clockwise order.

- An intersection region of disks is bounded by arcs.
- The arc $\Gamma_+(i,j), \Gamma_-(i,j)$ (counter-clockwise) determines a region where i and j intersect.
- In a counter-clockwise traversal, the arcs where $\Gamma_+(i,j) > \Gamma_-(i,j)$ can be a problem for the implementation. Work-around: repeat it.

The algorithm is described simply as:
For every disk, traverse the sorted list of intersection angles twice, keeping a set of active disks, and the current best solution.

Figure: The intersections list of a disk with three other disks.



The run-time complexity of the algorithm is $\mathcal{O}(n^2 \log n)$.

- There are $\mathcal{O}(n^2)$ intersections among n disks
- Sorting takes $\mathcal{O}(n^2 \log n)$
- The traversal takes $\mathcal{O}(n)$ for every disk.
- Can be implemented in $K \log n$ where K is the number of intersections.

Multiple disks

For every disk, try every possible assignment that the traversal goes through.

content... trabalhos passados

content... algoritmo

Trabalhos Futuros