Minimum Dilation Stars

David Eppstein and Kevin A. Wortman

Univ. of California, Irvine
Donald Bren School of Information and Computer Sciences

The Minimum Dilation Star Problem

Input: a set of n points (e.g., locations of airports)

Output: a point x (e.g. the hub of an airline network)

Dilation of points s, t for hub x
= factor by which travelers from s to t are inconvenienced
= ratio of path length |sx|+|xt| through x to direct path |st|

Choose x that minimizes the maximum dilation among all pairs

Constrained vs unconstrained

Unconstrained problem: x can be any point in the plane

Constrained problem: x must be one of the input points

Related work

Facility location:
choose center(s) for points optimizing some quality measure
typically involving point-center distances

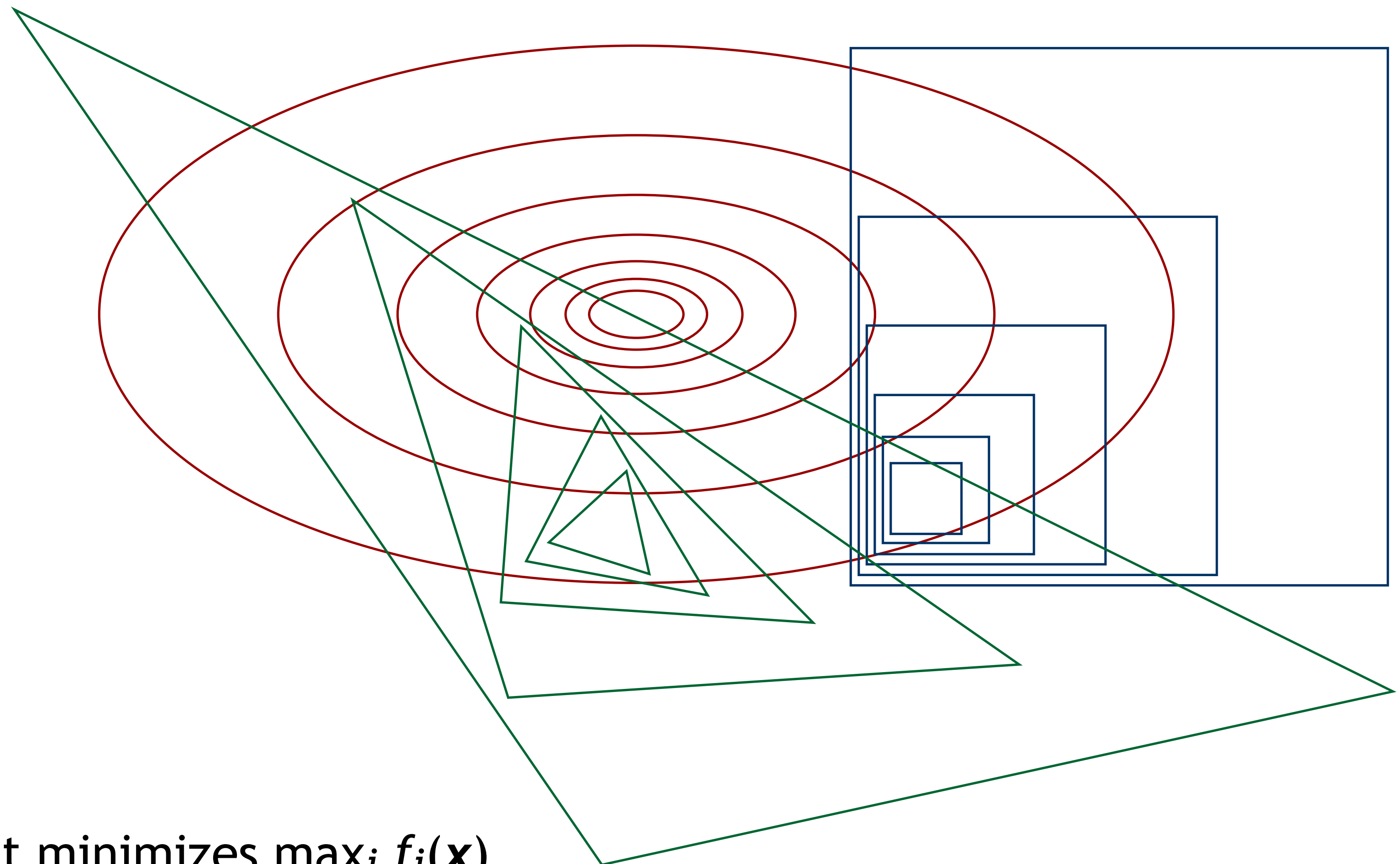
Network design:
Find optimal spanning tree or spanning network for a set of points

Spanner construction:
Find graph accurately representing distances in point set
(here, graph = star with input points as leaves)

All three have large literatures
but minimizing dilation appears to be novel

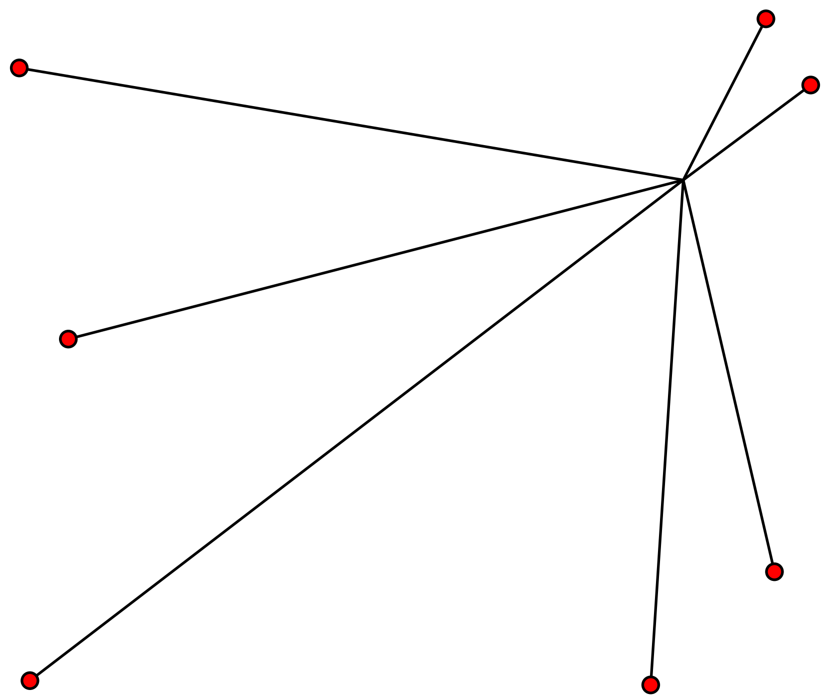
Quasiconvex programming [Amenta, Bern, Eppstein, J. Algorithms 1999, et seq.]

Input: family of quasiconvex functions $f_i(\boldsymbol{x})$, $\boldsymbol{x}$ in $\mathbf{R}^d$
i.e., lower level sets { $\boldsymbol{x}$ such that $f_i(\boldsymbol{x}) \leq L$ } are convex for all i, L



Output: $\boldsymbol{x}$ that minimizes $\max_i f_i(\boldsymbol{x})$

Unconstrained minimum dilation star as QCP



Input: point set S

Q(S) = set of $O(n^2)$ functions
(|sx|+|xt|)/|st|
measuring dilation of each input pair

Level sets are ellipses, so quasiconvex

Choose hub x minimizing
max { f(x) for f in Q(S) }

Leads to simple $O(n^2)$ algorithm: construct Q(S), apply QCP

Can we do better?

Implicit quasiconvex programming [Chan, SODA 2004]

Defined by a function Q mapping inputs to sets of quasiconvex functions

Q(Input) may be much larger than the input itself

Could solve by computing Q then applying any QCP algorithm

Chan showed many implicit QCPs can be solved more efficiently using a ***decision oracle*** and a ***subdivision process***

Decision oracle

Given implicit QCP input S, point x, value y
Is there a function f in Q(S) with f(x) > y?

Evaluation oracle

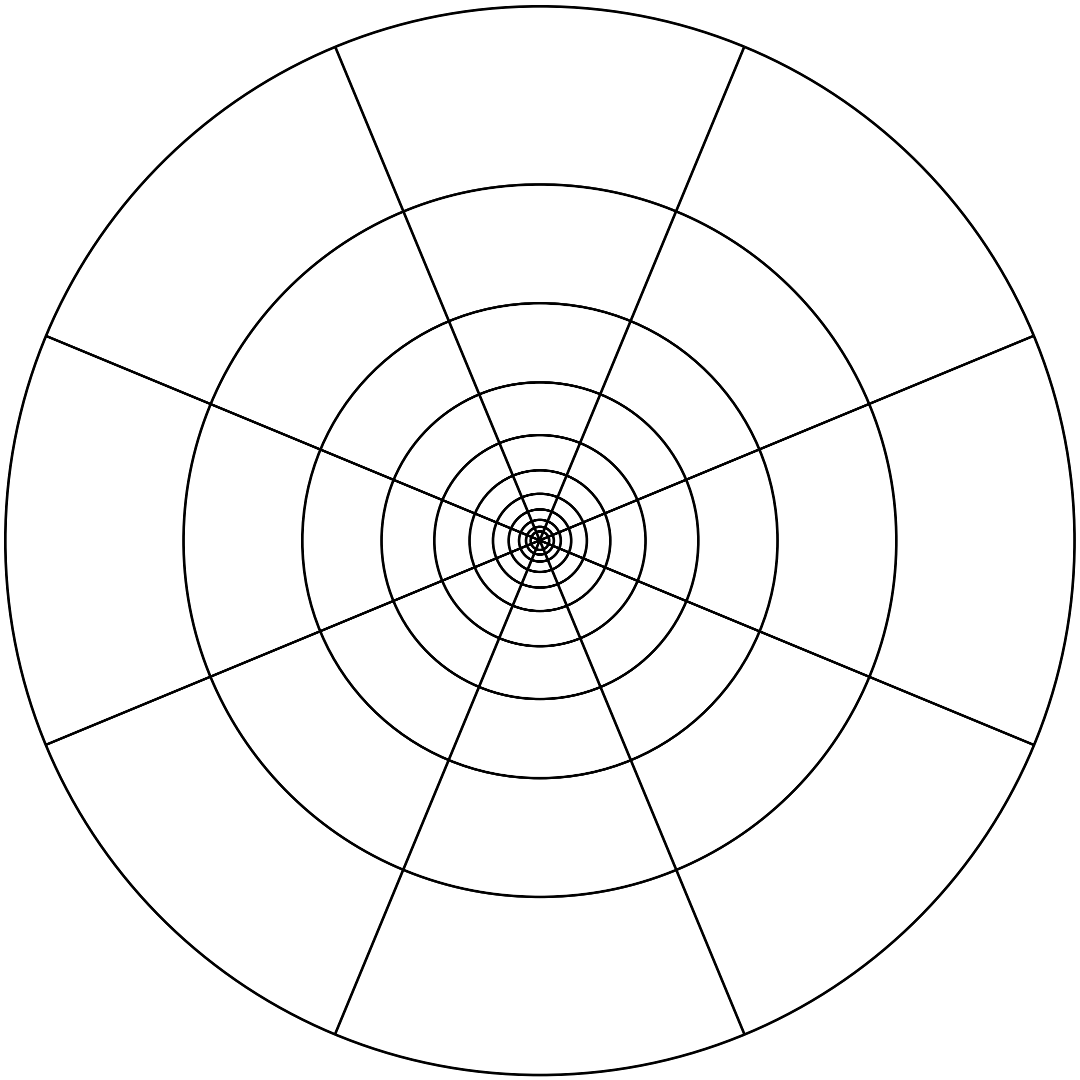
Given implicit QCP input S, point x, value y
Compute max f(x) among f in Q(S)

Decision from evaluation

If have evaluation oracle, decision is easy:
compute max f(x), compare to y

Evaluation oracle when max dilation is O(1):

Sort points by distance from x
Compute dilation from each point to O(1) near neighbors in sorted sequence
return maximum among computed dilation values



Why is this correct?

View points as partitioned into segments of annuli centered on x (conceptually, not in algorithm)

O(1) segments per annulus
Exponentially increasing radii (ratio depends on max dilation)

Each segment has O(1) points (else dilation would be too high)

The points s, t having max dilation must be in neighboring annuli (else dilation would be too low)

Evaluation oracle when max dilation is Omega(1):

Find O(1) nearest points to each input point
Compute dilation from each point to these O(1) nearby points
return maximum among computed dilation values

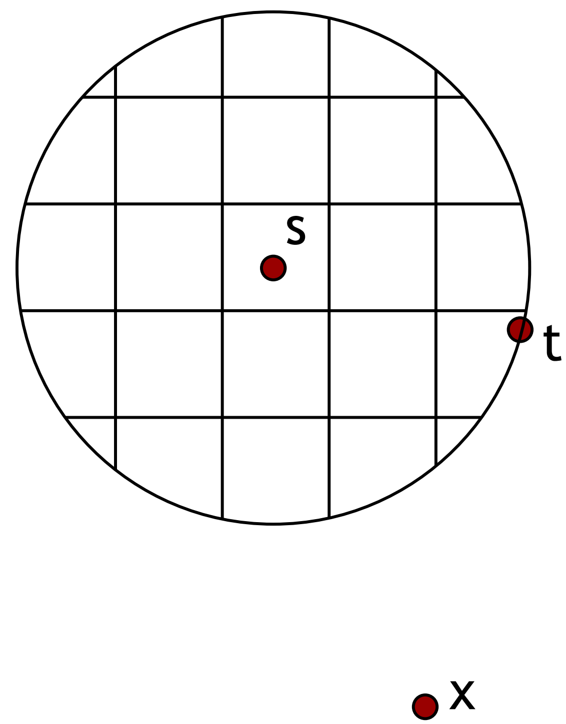
Why is this correct?

Let O = circle centered on s
with radius |st|

If dilation > 3, x is well outside O

Partition O into O(1) pieces
such that each piece has ≤ 1 point
(else dilation would be too high)

Number of points nearer to s than t
is bounded by the number of pieces



Subdivision process

For some constants $r = O(1)$, $a < 1$

Subdivide any input A into r smaller inputs A_0, A_1, A_2, ...

Each A_i is of size at most $a \bullet \text{size}(A)$

$Q(A)$ = union of $Q(A_i)$

Subdivision process for min dilation star

Each quasiconvex function is determined by some pair of points

Partition input S into three equal subsets S_0, S_1, S_2

Define subproblem $A_i = S \setminus S_i$

$r=3$, $a=2/3$

Chan's implicit QCP algorithm

Repeatedly subdivide into O(1) subproblems

Apply generalized linear programming

elements = subproblems
objective function = value of union of QCPs

GLP violation test = decision oracle
GLP basis change = recursive call to implicit QCP

Choose number of subproblems so that
E(total size of recursive calls) = constant fraction of input

Leads to randomized implicit QCP algorithm
expected time = O(decision oracle + subdivision process)

Result: O(n log n) time for unconstrained min dilation star

Constrained Problem Overview

To select best hub from a set H of candidates (initially all inputs):

repeat:

Pick a random point h from H

Evaluate the dilation D of h

Construct locus L of hubs with dilation < D

H = H intersect L

until H is empty; optimal hub is the last chosen h

Each iteration reduces candidates by a factor of 2 in expectation so O(log n) iterations

Bottleneck is construction of L

The locus of low-dilation hubs

L = { x such that max dilation of pairs (s,t) through x is < D }
= intersection of $O(n^2)$ similar ellipses, having each pair (s,t) as foci

But, only O(n) ellipses contribute to the intersection boundary!

O(n) having as foci a point and one of its k-nearest Euclidean neighbors
O(n) having as foci two points within O(1) positions of each other
in sorted order of distances from unconstrained center

(proof idea similar to unconstrained algorithm)

Can construct intersection of O(n) ellipses in time $O(n\ 2^{\mathrm{alpha}(n)} \log n)$

(standard application of Davenport-Schinzel sequence theory)

Outer sampling loop of constrained algorithm adds another log

Total time to find best hub among input points: $O(n\ 2^{\mathrm{alpha}(n)} \log^2 n)$

Conclusions

Unconstrained minimum dilation star:

O(n log n) expected time randomized algorithm

Works in any constant dimension

Constrained minimum dilation star:

$O(n\, 2^{\mathrm{alpha}(n)} \log^2 n)$ expected time randomized algorithm

Works only in the plane

Open: Derandomize? Higher dimension constrained problem?