Approximation Algorithms for Diameter, Width, and Related Problems

Timothy M. Chan

School of CS
Univ of Waterloo

INTRODUCTION

Why Approximation?

efficiency

e.g. dD width: $O(n^{\lceil d/2 \rceil}) \to O(n)$

allows for tougher versions

e.g. dynamic, kinetic, outliers, etc.

This Talk

focuses on just 2 specific problems
& on fixed d

Surveys a number of known alg'ms
& some new ones

+ previews results in "data stream" model

3 Strategies to Approx. Schemes

[from Schuurman, Woeginger's survey ~'07]

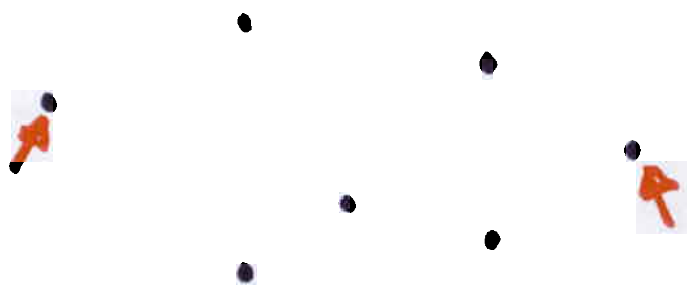
*1. structuring (e.g. rounding) the input

*2. structuring (") the output

3. structuring the execution of an alg'm

THE DIAMETER PROBLEM

given n pts P in $\mathbb{R}^d$,
find farthest pair



Quick Const-Factor Alg'ms

a) s = any pt
t = farthest neighbor of s

Approx factor: 2

Pf: $|pq| \le |ps| + |sq| \le 2|st|$ $\square$

b) [Egecioglu, Kalantari '89]

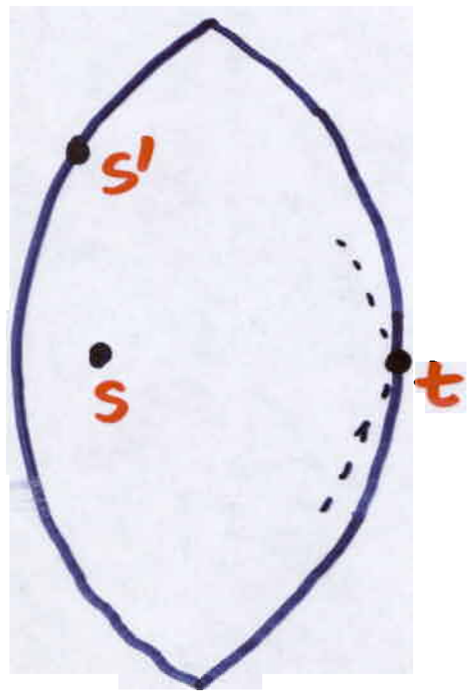
s = any pt

t = farthest neighbor of s

s' = farthest neighbor of t

Approx factor: $\sqrt{3}$

Pf:



draw lune with radius $r = |s't|$

$$\text{Diam}(P) \le \text{Diam}(\text{lune}) = \sqrt{3}\, r$$

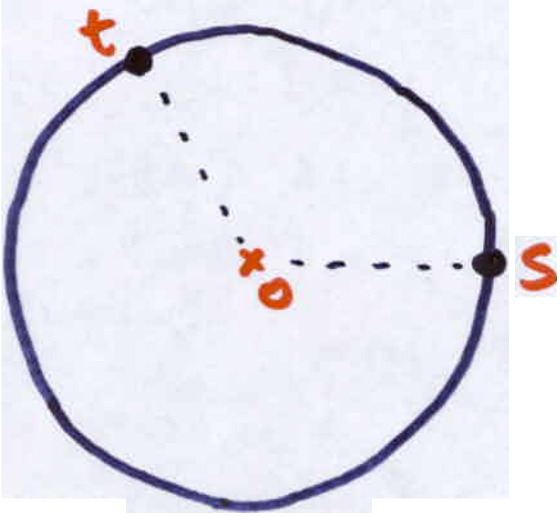
$\square$

c) [Goel, Indyk, Varadarajan '01]

$B =$ smallest enclos. ball (exact/approx)
w. center o, radius r

$s =$ a pt on ∂B

$t =$ a pt on ∂B w. $\angle sot \geq 90°$



Factor: $\sqrt{2}$ (roughly)

Pf: $\mathrm{Diam}(P) \leq 2r$

$|st| \geq \sqrt{2} r$ $\square$

Rounding the Input (i.e. Points)

[Barequet, Har-Peled '99]

1. D = const-factor approx.
2. round pts to grid of side εD
3. return diameter of rounded pts

Approx factor: $1 + O(\varepsilon)$

Pf: additive error is $O(\varepsilon D)$. $\square$

Runtime: 2. gives $O(1/\varepsilon^d)$ pts

3. by brute force

$\Rightarrow \boxed{O(n + 1/\varepsilon^{2d})}$

Rmks: can improve to $O(n + 1/\varepsilon^{2(d-1)})$

idea works for k-center, MDST, etc.

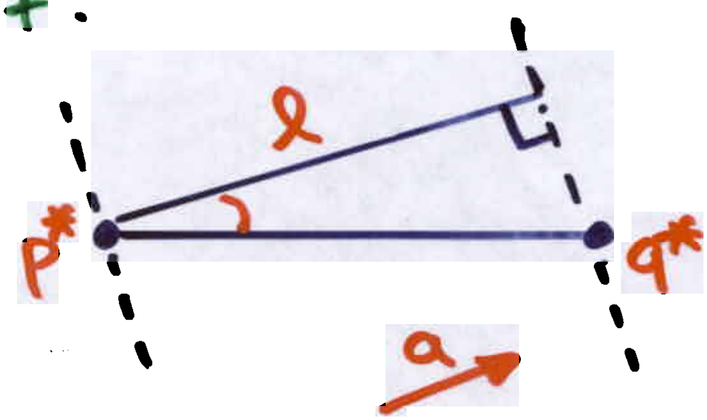
Rounding "Output" (i.e. Direction)

[Agarwal, Matoušek, Suri '92]

1. form $O(1/\delta^{d-1})$ vectors A s.t.
   $\forall x, \exists a \in A, \angle(a,x) = O(\delta)$
2. find extreme pt pair $p_a q_a$
   along each direction $a \in A$

Factor: $1 + O(\delta^2)$ (set $\delta = \sqrt{\varepsilon}$)

Pf:



let $a \in A$

$\angle(a, p^*q^*) = O(\delta)$

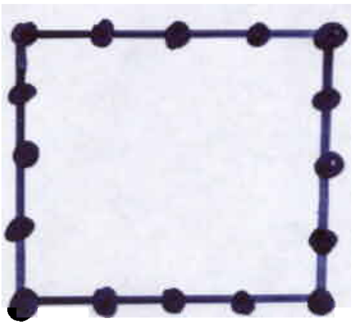
$$|p_a q_a| \geq \ell = |p^*q^*| / \cos O(\delta) \quad \square$$

Runtime: $\boxed{O(n/\varepsilon^{(d-1)/2})}$

Options for the vectors A :

- pack pts on unit sphere (e.g. Yao '82)

- or grid pts on unit hypercube's bdry



(e.g. Agarwal, Har-Peled '01)

Rmks :

- idea works for nearest neighbors, MST, etc.
- can slightly improve runtime by data structures for extreme pts

Combo: Rounding Pts AND Directions

[Chan '00]

Runtime: $O(n + 1/\varepsilon^{d-1} + n'/\varepsilon^{(d-1)/2})$

with $n' = O(1/\varepsilon^{d-1})$

$\Rightarrow \boxed{O(n + 1/\varepsilon^{3(d-1)/2})}$

More Alg'ms [Chan '00] (skip)

2 ways to get runtime near

$O(n + 1/\varepsilon^{d})$

idea - recurse in dimension ...

DIGRESSION : "Discrete Voronoi Diagrams"

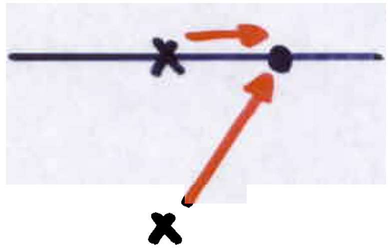
given n grid pts in $\{1, \ldots, E\}^d$,
find nearest neighbor (NN) of
every grid pt in $\{1, \ldots, E\}^d$

Rmks:

- different from "approx. VDs"
- image processing appls
- studied by Breu, Gil, Kirkpatrick, Werman'95

An Alg'm

idea - recurse in dimension



Rmk: works for farthest neighbors,
or extreme pts along grid directions

BACK TO DIAMETER . . .

New (Final?) Alg'm

just run Combo
but use above method to find
the extreme pts

$\Rightarrow \boxed{O(n + 1/\varepsilon^d)}$ time

(or more precisely, $O(n + 1/\varepsilon^{d-(3/2)})$)

Open: $O(n + 1/\varepsilon^{(d-1)/2})$ time ??

comparison w. practical methods ?
[Har-Peled '02, Malandain, Boissonat '02]

To find NN of every pt in a kD grid:

1. solve problem for pts in each horizontal slice of G

2. for each vertical line ℓ of G,

   let $p_1, \dots, p_E$ be the NNs from 1.

   construct the 1D VD of $p_1, \dots, p_E$ at ℓ

   look up answers

Runtime:

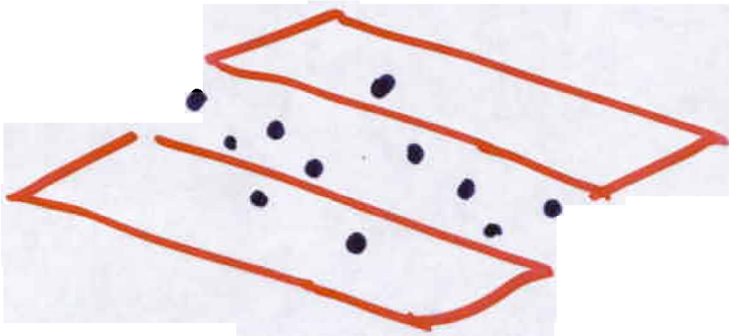
1D VD reduces to 2D CH, computable by Graham scan in $O(E)$ time

$$T_k(n) = \sum_{i=1}^{E} T_{k-1}(n_i) + O(E^{k-1} \times E)$$

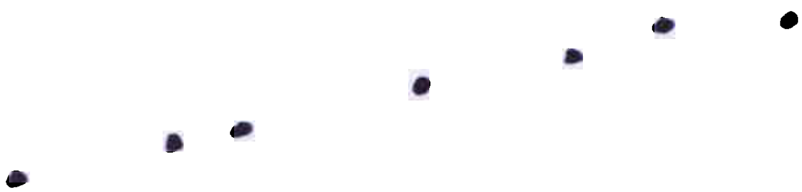
$$\Rightarrow \quad T_d(n) = \boxed{O(E^d)}$$

THE WIDTH PROBLEM

given n pts P in $\mathbb{R}^d$,
find 2 parallel hyperplanes
enclos. P, with min distance



Rounding Pts: doesn't seem to work ...



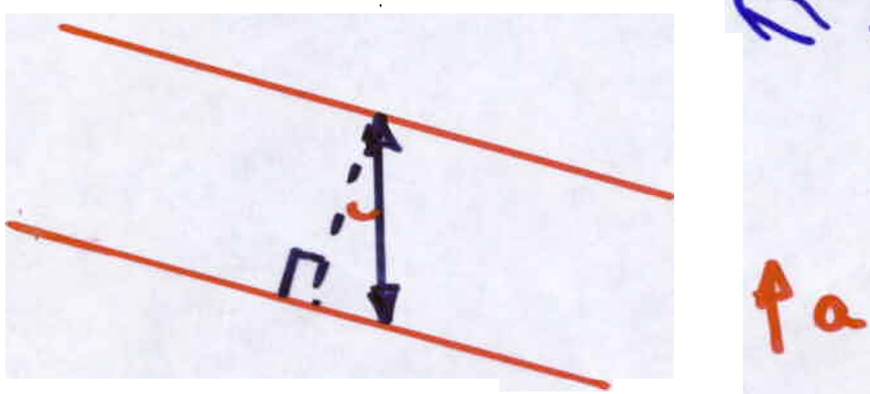
Rounding Directions: not work either?

First Alg'm [Duncan, Goodrich, Ramos '97]

1. form vectors A as before
2. for each $a \in A$,

   find 2 parallel hyperplanes (any direction) enclos. P, minimizing distance along direction a

this is LP



Factor: $1 + O(\delta^2)$ (set $\delta = \sqrt{\varepsilon}$)

Pf: let $a \in A$, $\angle(a, a^*) = O(\delta)$

dist along $a \leq$ dist along $a^* / \cos O(\delta)$ $\square$

Runtime: $\boxed{O(n/\varepsilon^{(d-1)/2})}$

(or slightly better by data structures for LP)

Rounding Pts (1st version)

1. let s, t, W be as before
2. draw grid lines parallel to $\overleftrightarrow{st}$ at distance $\varepsilon W, 2\varepsilon W, \ldots, W$
3. round pts to lie on grid lines
4. return width of rounded pts

Factor: $1 + O(\varepsilon)$

Pf: additive error is $O(\varepsilon W)$ $\square$

Runtime: 2. gives $O(1/\varepsilon)$ pts

$\Rightarrow O(n + 1/\varepsilon)$

Rounding Pts (2nd version)

1. let s, t be as before, $D = |st|$
2. draw grid lines orthogonal to $\overrightarrow{st}$ at distance $\varepsilon D, 2\varepsilon D, \ldots, D$

3, 4. as before

Factor: $1 + O(\varepsilon)$

Pf: for each direction a,
width along a changes by
$O(\varepsilon \cdot \text{width of } \overline{st} \text{ along } a)$
$\leq O(\varepsilon \cdot \text{width of } P \text{ along } a)$ $\square$

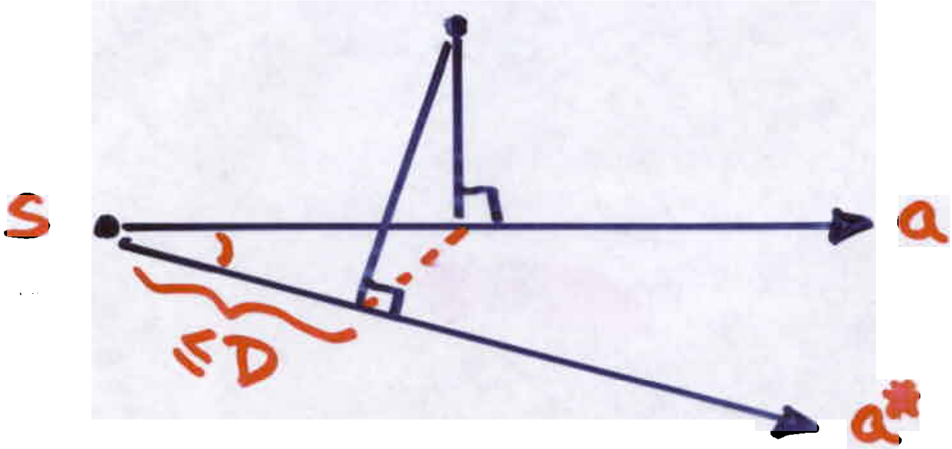
Runtime: $O(n + 1/\varepsilon)$

Rounding Directions [Agarwal, Aronov, Sharir '97]

1. let s, t, W, D be as before
2. find width along directions at angle $\varepsilon W/D$, $2\varepsilon W/D$, ..., $O(W/D)$ with normal of $\overline{st}$

Factor: $1 + O(\varepsilon)$

Pf: say $\angle(a, a^*) = O(\delta)$



$$|\text{width along } a^* - \text{width along } a| \leq O(D \cdot \varepsilon W/D) = O(\varepsilon W) \quad \square$$

Runtime: $O(n/\varepsilon)$

Moral

rounding can work, but need grid over boxes (rotated) instead of hypercubes

Generalization to dD

[Agarwal, Har-Peled '01] (sketch)

take a bounding box [Barequet, Har-Peled '9
rotate & re-scale it into hypercube
now use a standard grid approach
(round pts and/or directions)

Runtime: $O(n + 1/\varepsilon^{2(d-1)})$

or by a combo, $\boxed{O(n + 1/\varepsilon^{3(d-1)/2})}$

New Result

run Agarwal, Har-Peled's alg'm
with our discrete VD method

$\Rightarrow \boxed{O(n + 1/\varepsilon^{d-1})}$ time

Rmks:

- AH's alg'm approximates width along all directions ("extent fn")
- leads to "core sets" for min enclos. cylinder / annulus / cylind. shell, kinetic problems, etc.
  [Har-Peled, Varadarajan '01]

DATA STREAM COMPUTATION

The Model

goal - handle massive data sets

alg'm can make just one pass over input

space is limited (can't store whole input)

Example: Diameter

recall alg'm by rounding direction

$O(1/\varepsilon^{(d-1)/2})$ space & time per pt

Let's Try 2D Width ...

traditional alg'ms, e.g. Janardan '93

$O(n)$ space, $O((\log n)/\sqrt{\varepsilon})$ time per pt

Agarwal, Har-Peled, Varadarajan '03

$O((\log^2 n)/\sqrt{\varepsilon})$ space & time

Hershberger, Suri [MPDS '03]

(needs bounded ratio W/D)

Question: const space?

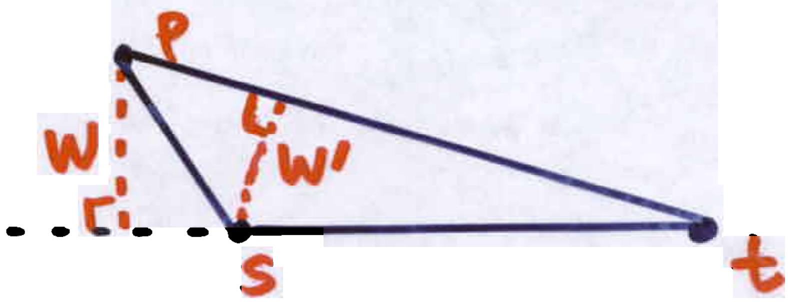
Let's Start Over in 2D . . .

A Const-Factor Alg'm [Agarwal, Aronov, Sharir '97]

s = any pt
t = farthest neighbor of s
w = farthest distance to $\overleftrightarrow{st}$

Factor: 4

Pf: $\text{Width}(P) \le 2w$



$$w\,|st| = w'\,|pt|$$
$$|pt| \le 2|st| \Rightarrow w' \ge w/2$$

$$\text{Width}(P) \ge \text{Width}(\{s,t,p\})$$
$$\ge w/2 \qquad \square$$

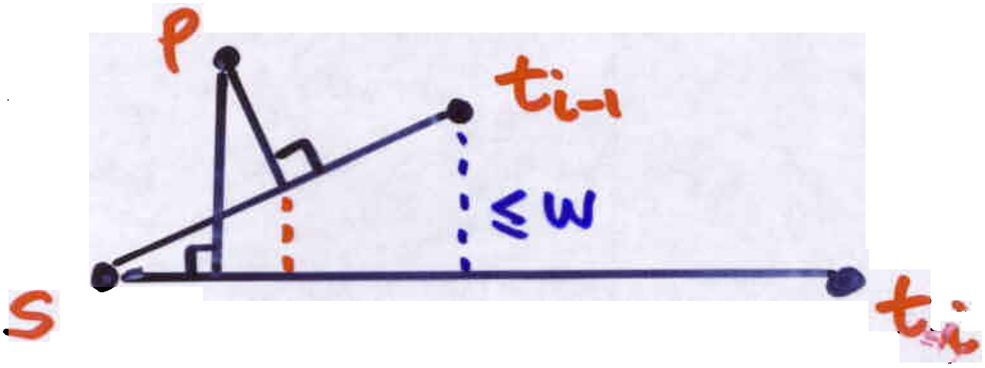
Approx factor: $O(1)$

Pf: $\text{Width}(P) \geq W/c$ as before

let t_i = i^{th} version of t since p is processed

know $|st_i| \geq 2|st_{i-1}|$

$|sp| \leq 2|st_1|$



$$d(p, \overleftrightarrow{st_i}) - d(p, \overleftrightarrow{st_{i-1}}) \leq \frac{|sp|}{|st_{i-1}|} W$$

$$\Rightarrow d(p, \overleftrightarrow{st_\infty}) - d(p, \overleftrightarrow{st_1}) \leq \left(2 + 1 + \tfrac{1}{2} + \tfrac{1}{4} + \cdots\right) W$$

$$\Rightarrow \text{Width}(P) \leq c' W$$

$\square$

A New, One-Pass, Const-Factor Alg'm !

s, t = first two pts

for each $p \in P$,

  if $|sp| \leq 2\,|st|$

    $w = \max\{w, d(p, \overleftrightarrow{st})\}$

  else $w = \max\{w, d(t, \overleftrightarrow{sp})\}$

    $t = p$

Space: $O(1)$

Refinement & Generalization to dD (skip)

Space (& Time): $\boxed{O\left(\left(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon}\right)^{d-1}\right)}$

Rmk:

works for the core sets of
Agarwal, Har-Peled, Varadarajan
($\Rightarrow$ min enclos ball, annulus, etc.)

CONCLUSION

diameter & width are good "test cases" for geom. approx. techniques

Open: e.g. best dynamic data structures?