

Dynamic Coresets

Timothy Chan

School of CS

U of Waterloo

The Coreset Concept

- To approximately solve a problem for a point set:
 - Find a **good subset** S of small (const) size
 - Return sol'n for S

The Coreset "Phenomenon"

- Agarwal, Har-Peled, Varadarajan [SODA'01/FOCS'01]
(const dim: diameter, width, min enclos. cylinder, min enclos. box, min enclos. annulus, cylindrical shell,; moving pts...)

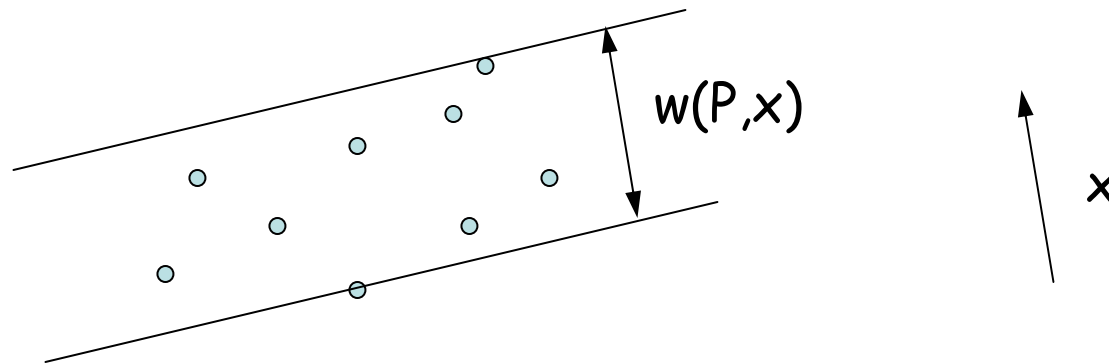
The Coreset "Phenomenon"

- Agarwal, Har-Peled, Varadarajan [SODA'01/FOCS'01]
(const dim: diameter, width, min enclos. cylinder, min enclos. box, min enclos. annulus, cylindrical shell,; moving pts...)
- Har-Peled, Wang [SoCG'03], Agarwal, Har-Peled, Yu [SODA'06]
(const dim: ... with outliers)
- Bădiou, Har-Peled, Indyk [STOC'02], Bădiou, Clarkson [SODA'03]
(high dim: min enclos. ball, k-center, ...)
- Har-Peled, Varadarajan [SoCG'02]
(high dim: 1-[j-flat]-center, ...)
- Har-Peled, Mazumdar [STOC'04], Har-Peled, Kushal [SoCG'05]
(const dim: k-median, k-means, ...)
- Etc. etc. etc.

Our Focus: Coreset for Extent

[Agarwal, Har-Peled, Varadarajan]

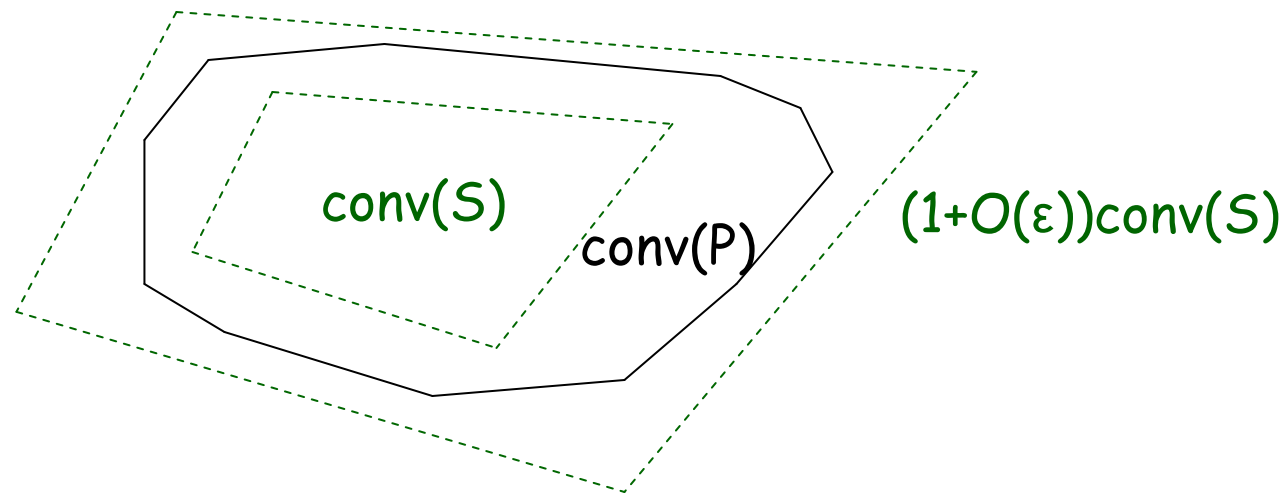
- **Def:** Given point set P in \mathbb{R}^d for const d ,
 $w(P, x)$ = width of slab enclosing P
orthogonal to direction x



- **Def:** $S \subset P$ is an **extent coreset** of P if
 $w(S, x) \geq (1-\epsilon) w(P, x)$ for all x

Why Coreset for Extent?

- "Approximate convex hull" in a strong sense...



- In particular: $(1+O(\epsilon))$ -approximates diameter, width, min enclos. cylinder, min enclos. box, min enclos. annulus, cylindrical shell,

Known Alg'ms

- Extent coresnet of size $O(\varepsilon^{-(d-1)/2})$
 - Agarwal, Har-Peled, Varadarajan
time roughly $O(n + \varepsilon^{-1.5d})$
 - Chan [SoCG'04]
time roughly $O(n + \varepsilon^{-d})$

What about Dynamic Alg'ms?

- Maintain sol'n under **insert** & **delete**?

Known Dynamic Alg'ms

- Exact convex hulls:

d=2: Overmars, van Leeuwen'80	$O(\log^2 n)$ time
...	
Chan'99	$O(\log^{1+\varepsilon} n)$
Brodal, Jacob'00	$O(\log n \log \log n)$
Brodal, Jacob'02	$O(\log n)$
d=3: easy	$O^*(n^{1/2})$
Agarwal, Matoušek'93	$O(n^\varepsilon)$
Chan [SODA'06]	$O(\log^6 n)$
d=4: most likely no polylog alg'm...	

Known Dynamic Alg'ms

- Approx. width for $d=2$:

Janardan'93, Rote, Schwarz, Snoeyink'93 \Rightarrow dynamic convex hull

- Extent coresset for any const d :

Agarwal, Har-Peled, Varadarajan $O(\log^{O(d)} n)$
(e.g., for $d=3$: $O(\log^2 n \log \log n)$)

- Insert-only/"streaming":

Agarwal, Har-Peled, Varadarajan $O(\log^{O(d)} n)$

Chan [SoCG'04] $O(1)$

- Streaming w. insert & delete:

Frahling, Sohler'05 $O(\log^{O(d)} U)$

Sample Open Problem

- Dynamic approx. width for $d=3$ in $O(\log n)$ time?

New Result

- Dynamic approx. width for $d=3$ in $O(\log n)$ time? **YES!!**

New Result

- Dynamic approx. width for $d=3$ in $O(\log n)$ time? **YES!!**
- Dynamic approx. width **for any const d** in $O(\log n)$ time !!

New Result

- Dynamic approx. width for $d=3$ in $O(\log n)$ time? **YES!!**
- Dynamic approx. width **for any const d** in $O(\log n)$ time !!
- Dynamic approx. min enclos. cylinder, min enclos. box, min enclos. annulus, for any const d , all in $O(\log n)$ time !!!

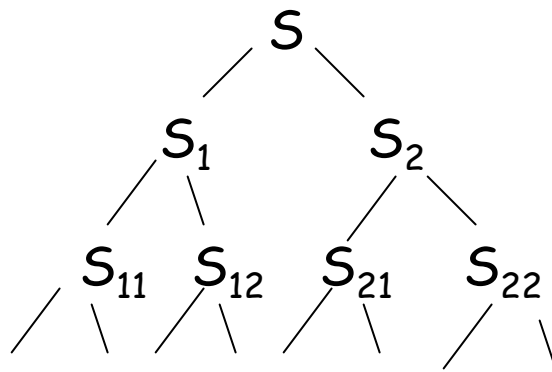
New Result

- Dynamic extent coresets for any d in $O(\log n)$ time

Previous Dynamic Coreset Alg'm

[Agarwal, Har-Peled, Varadarajan]

- Binary tree of coresets of unions of coresets of



coresets are
"decomposable"

⇒ update time $O(\epsilon'^{-O(d)} \log n)$

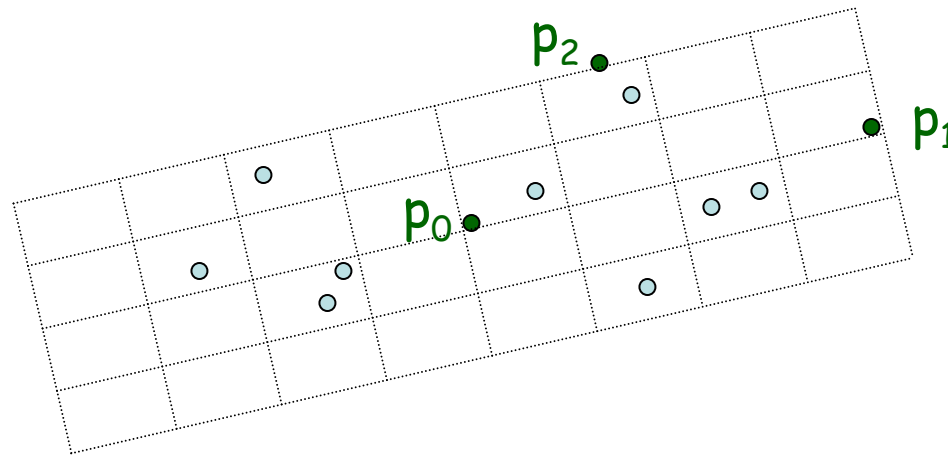
- But need $\epsilon' := \epsilon / \log n \Rightarrow O(\log^{O(d)} n)$

A Previous Static Coreset Alg'm

[Agarwal, Har-Peled, Varadarajan]

- Take a pt p_0
- Take farthest pt p_1 from p_0
- Take farthest pt p_2 from $\text{line}(p_0p_1)$
- Take farthest pt p_3 from $\text{plane}(p_0p_1p_2)$

.....



- Use "anchor pts" p_0, \dots, p_d to form **grid** of $O(\epsilon^{-d})$ cells
- Keep ≤ 1 pt per cell

A New, Simple Dynamic Coreset Alg'm

- Idea: randomization + "dynamic prune & search"
- Take a random pt p_0
Let $A_1 =$ the $n/(2d)$ farthest pts from p_0
Take a random pt p_1 from A_1 ; remove all pts of A_1
Let $A_2 =$ the $n/(2d)$ farthest pts from line(p_0p_1)
Take a random pt p_2 from A_2 ; remove all pts of A_2
.....
- Use these "anchor pts" p_0, \dots, p_d to form grid
- Recurse on the $\leq n/2$ removed pts !!

The New Alg'm (cont'd)

- To delete x :
 - if x is not an anchor pt then easy
 - else rebuild [prob. of this case is $O(1/n)$]
- Also rebuild after every δn updates

- Preprocessing time:

$$P(n) = P(n/2) + O(n) \Rightarrow O(n)$$

- Expected amort. update time:

$$U(n) = U(n/2) + O(n) \cdot O(1/n) \Rightarrow O(\log n)$$

Further Results

- **Deterministic** $O(\log n)$ update time
[idea: derandomize by "approx centerpts"]
- **Coreset for k-center:**
 $O(k^{O(1)} \log n)$ time (rand.)
- Other problems?

Is $O(\log n)$ Optimal?

Is $O(\log n)$ Optimal?

- Integer input on the word RAM:
 $O(\log \log U)$ time
[idea: combine a straightforward $O(\log^{O(d)} U)$ alg'm with the $O(\log n)$ alg'm]
- Const-factor approx width for $d=2$:
 $O(1)$ time
- Open: $O(1)$ for extent coresets??