Approximating Extent Measures of Points

Pankaj K. Agarwal
Sariel Har-Peled
Kasturi R. Varadarajan

CS468, Winter 2006

Extent Measures

- Given a point set $P \subseteq \mathbb{R}^d$
- **Extent measures** – statistics about P or its enclosing shape
- Some examples
  - k-th largest distance
  - Min. volume bounding box
  - Min. width bounding slab
  - Min. enclosing sphere
  - Min. enclosing cylinder
  - Min. width enclosing spherical shell
  - Min. width enclosing cylindrical shell

Main Result

- **Technique for ϵ-approximating (a large class of) extent measures**

- Compute a **subset of the input** $Q \subseteq P$ (**coreset**) which
  - Preserves the solution to ϵ-accuracy
  - Small size (does not depend on $|P|$, only on ϵ)

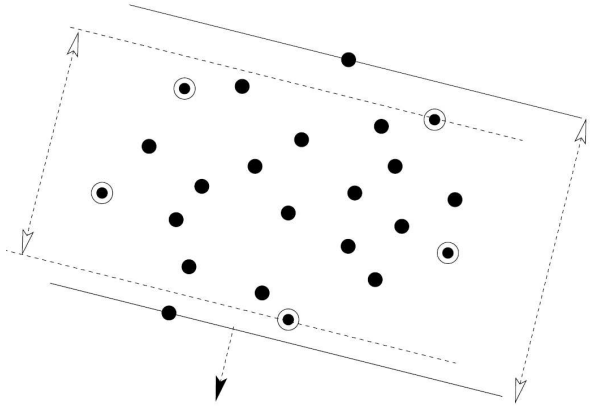
- **General properties**
  - Strong LTAS, running time $O\left(n + \left(\frac{1}{\epsilon}\right)^{O(1)}\right)$
  - Coreset size $O\left(\left(\frac{1}{\epsilon}\right)^{O(1)}\right)$
  - Exponents depend on d

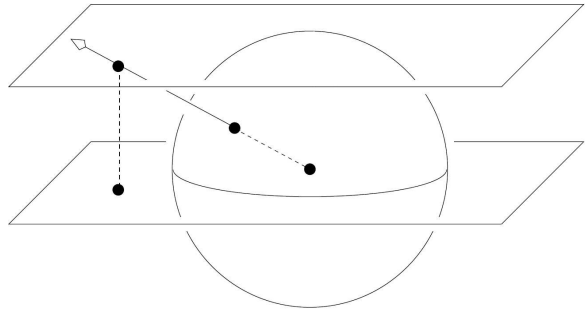
- Simple to implement, and some improvements
  - Min. enclosing spherical shell
    exponent down from $O(d^2)$ [Chan 02] to $O(d)$

Key Definitions

- Lead to two main approximation primitives
- **Directional width** of a point set P in the direction $u \in \mathbb{R}^{d-1}$

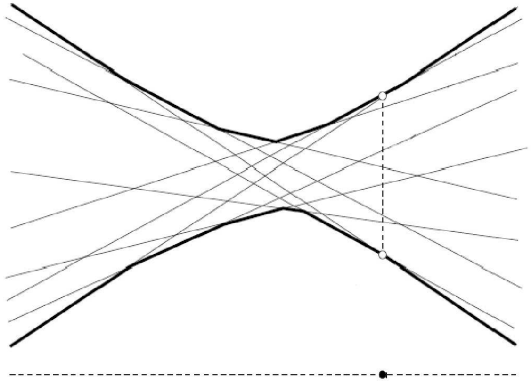
$$w(u, P) = \max_{p \in P} \langle [u, 1], p \rangle - \min_{p \in P} \langle [u, 1], p \rangle$$





- **Extent** of a set F of $(d-1)$-variate functions at $x \in \mathbb{R}^{d-1}$

$$e(x, F) = \max_{f \in F} f(x) - \min_{f \in F} f(x)$$



Optimization Primitives

- $Q \subseteq P$ is an ϵ-**approximation** for P on $\Delta \subseteq \mathbb{R}^{d-1}$ if for all $u \in \Delta$

$$(1-\epsilon)w(u,P) \leq w(u,Q) \leq w(u,P)$$

- $G \subseteq F$ is an ϵ-**approximation** for F on $\Delta \subseteq \mathbb{R}^{d-1}$ if for all $x \in \Delta$

$$(1-\epsilon)e(x,F) \leq w(u,G) \leq e(x,F)$$

- **Note:** Always pick a subset of the input – **coreset**

Classes of Extent Measures

- **Faithful**
  - Approximated through directional width
  - "Convex" measures (bounding shapes)
- **Other**
  - Approximated through extent
  - "Concave" measures ("shells")

Overview

(A) **Strong LTASs for directional width**

- Reduction to "fat" point sets
- Algorithm 1: Grid
- Algorithm 2: Polytope
- Algorithm 3: Decomposition

Strong LTASs for extent

- Linear functions (hyperplanes)
- Polynomial functions
- r-th roots of polynomials

(B) **Dynamic updates**

Applications to specific extent measures

Reduction to "Fat" Point Sets

- A point set $P \in \mathbb{R}^d$ is α-**fat** if there exists a translation t such that
$$\alpha\mathbb{C} \subseteq \mathcal{CH}(P) + t \subseteq \mathbb{C} = [-1, 1]^d$$

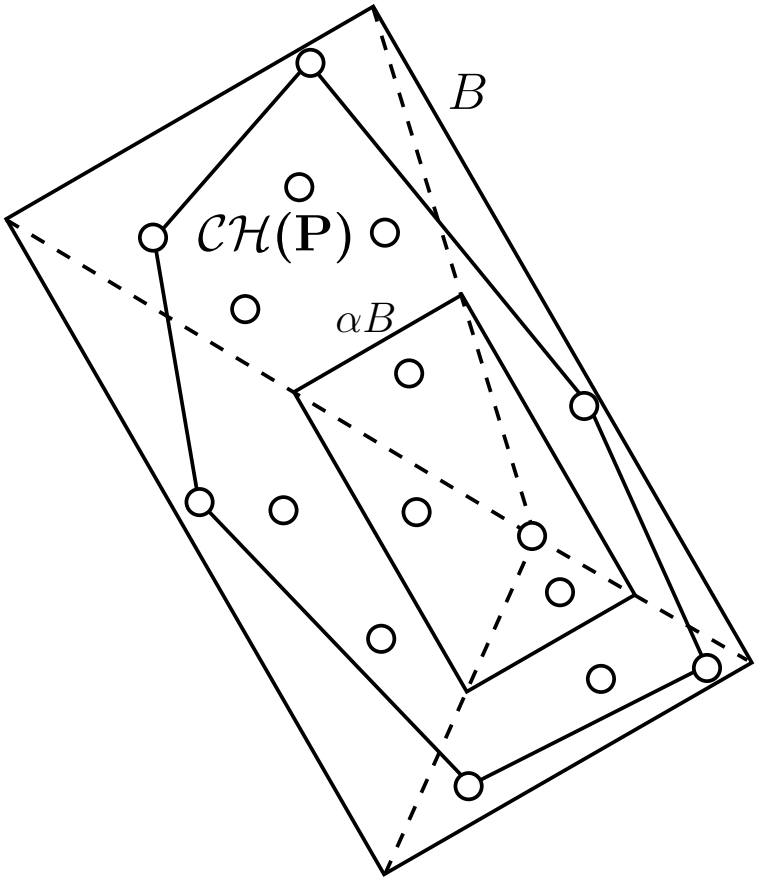
- **Sufficient to consider computing coresets for α-fat point sets**

- **Step 1:**
  Every point set can be made α-fat by applying a linear transformation, where $\alpha = \alpha(d)$
- **Step 2:**
  Every linear transformation preserves the approximation ratio of an arbitrary coreset
  - The size and construction time are clearly preserved

Step 1: There Exists a "Fattening Transform"

- [Barequet, Har-Peled 01]: Let $P \subseteq \mathbb{R}^d$ be of size n. Can compute in $O(n)$ time a box B and a vector $t \in \mathbb{R}^d$ such that

$$\alpha B \subseteq \mathcal{CH}(P + t) \subseteq B$$



- Recall: $\alpha = 1/10$? for $d = 3$
- Choose T so that $T(B) = \mathbb{C}$

Step 2: Invariance Under Linear Transforms

- **Lemma:**
  Let $T(x) = Mx + b$ be a non-degenerate linear transform.

$$Q \subseteq P \text{ } \epsilon\text{-approximates } P \text{ over } \Delta \subseteq \mathbb{R}^{d-1}$$

if and only if

$$T(Q) \text{ } \epsilon\text{-approximates } T(P) \text{ over}$$
$$\{v \mid [v, 1] = M^T[u, 1], \ u \in \Delta\}$$

- **Proof:** Easy by definition
  - Note: can assume $T(x) = Mx$
  - Also, $\langle [u, 1], Mp \rangle = \langle M^T[u, 1], p \rangle$

The Case of α-fat Point Sets

- From now on assume $\alpha\mathbb{C} \subseteq \mathcal{CH}(P) \subseteq \mathbb{C}$ where α depends only on d, not on n

The Case of α-fat Point Sets

- From now on assume $\alpha\mathbb{C} \subseteq \mathcal{CH}(P) \subseteq \mathbb{C}$ where α depends only on d, not on n
- **Lemma** [Rough approximation]: $w(x, P) \geq 2\alpha||x||$

  $\forall x \in \mathbb{R}^d$: $\max_{p\in P}\langle x, p\rangle = ||x|| \cdot \max_{p\in P} \underbrace{\langle \frac{x}{||x||}, p\rangle}_{\text{PROJECTION}} \geq ||x||\alpha$

  For $x_d = 1$: $\max\limits_{p\in P}\langle x, p\rangle \geq \alpha||x||$ $\min\limits_{p\in P}\langle x, p\rangle \leq -\alpha||x||$

The Case of α-fat Point Sets

- From now on assume $\alpha\mathbb{C} \subseteq \mathcal{CH}(P) \subseteq \mathbb{C}$ where α depends only on d, not on n
- **Lemma** [Rough approximation]: $w(x, P) \geq 2\alpha||x||$

  $\forall x \in \mathbb{R}^d$: $\max_{p\in P}\langle x, p\rangle = ||x|| \cdot \max_{p\in P} \underbrace{\langle \frac{x}{||x||}, p\rangle}_{\text{PROJECTION}} \geq ||x||\alpha$

  For $x_d = 1$: $\max\limits_{p\in P}\langle x, p\rangle \geq \alpha||x||$ $\min\limits_{p\in P}\langle x, p\rangle \leq -\alpha||x||$
- **Lemma** [Hausdorff dist.]: If $\max_{p\in P} \min_{q\in Q} ||p - q|| \leq \epsilon\alpha$ then Q is an ϵ-approximation for P

The Case of α-fat Point Sets

- From now on assume $\alpha\mathbb{C} \subseteq \mathcal{CH}(P) \subseteq \mathbb{C}$ where α depends only on d, not on n
- **Lemma** [Rough approximation]: $w(x, P) \geq 2\alpha||x||$

  $\forall x \in \mathbb{R}^d$: $\max_{p\in P}\langle x, p\rangle = ||x|| \cdot \max_{p\in P} \underbrace{\langle \frac{x}{||x||}, p\rangle}_{\text{PROJECTION}} \geq ||x||\alpha$

  For $x_d = 1$: $\max\limits_{p\in P}\langle x, p\rangle \geq \alpha||x||$ $\min\limits_{p\in P}\langle x, p\rangle \leq -\alpha||x||$
- **Lemma** [Hausdorff dist.]: If $\max_{p\in P} \min_{q\in Q} ||p - q|| \leq \epsilon\alpha$ then Q is an ϵ-approximation for P

$$w(x, P) - w(x, Q) \quad \leq \quad \langle x, p_1 - p_2\rangle - \langle x, q_1 - q_2\rangle$$

The Case of α-fat Point Sets

- From now on assume $\alpha\mathbb{C} \subseteq \mathcal{CH}(P) \subseteq \mathbb{C}$ where α depends only on d, not on n
- **Lemma** [Rough approximation]: $w(x, P) \geq 2\alpha||x||$

  $\forall x \in \mathbb{R}^d$: $\max_{p\in P}\langle x, p\rangle = ||x|| \cdot \max_{p\in P} \underbrace{\langle \frac{x}{||x||}, p\rangle}_{\text{PROJECTION}} \geq ||x||\alpha$

  For $x_d = 1$: $\max\limits_{p\in P}\langle x, p\rangle \geq \alpha||x||$ $\min\limits_{p\in P}\langle x, p\rangle \leq -\alpha||x||$
- **Lemma** [Hausdorff dist.]: If $\max_{p\in P} \min_{q\in Q} ||p - q|| \leq \epsilon\alpha$ then Q is an ϵ-approximation for P

$$\begin{aligned} w(x, P) - w(x, Q) &\leq \langle x, p_1 - p_2\rangle - \langle x, q_1 - q_2\rangle \\ &\leq |\langle x, p_1 - p_2\rangle| - |\langle x, q_1 - q_2\rangle| \end{aligned}$$

The Case of α-fat Point Sets

- From now on assume $\alpha\mathbb{C} \subseteq \mathcal{CH}(P) \subseteq \mathbb{C}$ where α depends only on d, not on n
- **Lemma** [Rough approximation]: $w(x, P) \geq 2\alpha||x||$

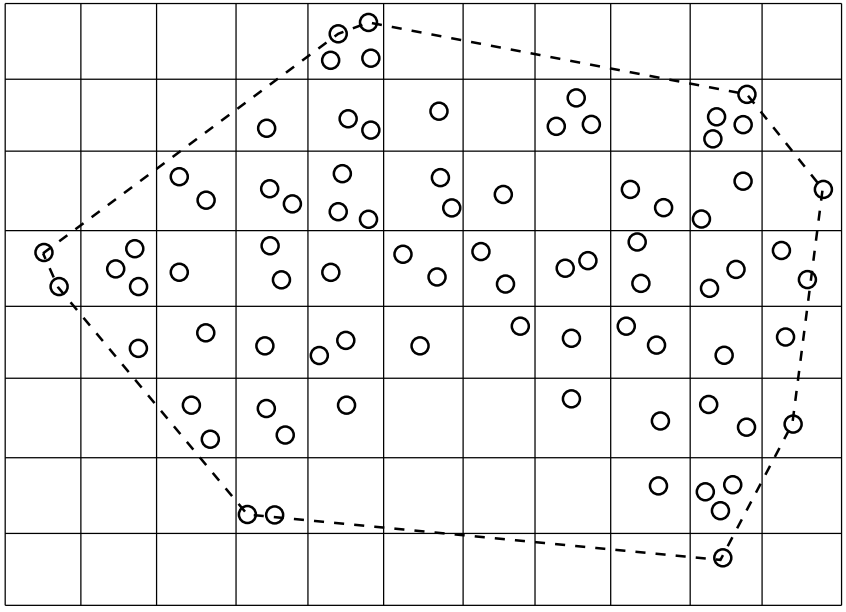
  $\forall x \in \mathbb{R}^d$: $\max_{p\in P}\langle x, p\rangle = ||x|| \cdot \max_{p\in P} \underbrace{\langle \frac{x}{||x||}, p\rangle}_{\text{PROJECTION}} \geq ||x||\alpha$

  For $x_d = 1$: $\max\limits_{p\in P}\langle x, p\rangle \geq \alpha||x||$ $\min\limits_{p\in P}\langle x, p\rangle \leq -\alpha||x||$
- **Lemma** [Hausdorff dist.]: If $\max_{p\in P} \min_{q\in Q} ||p - q|| \leq \epsilon\alpha$ then Q is an ϵ-approximation for P

$$
\begin{aligned}
w(x, P) - w(x, Q) \quad &\leq \quad \langle x, p_1 - p_2\rangle - \langle x, q_1 - q_2\rangle \\
&\leq \quad |\langle x, p_1 - p_2\rangle| - |\langle x, q_1 - q_2\rangle| \\
&\leq \quad |\langle x, (p_1 - q_1) - (p_2 - q_2)\rangle| \quad \leq \quad ||x|| \cdot 2\alpha\epsilon
\end{aligned}
$$

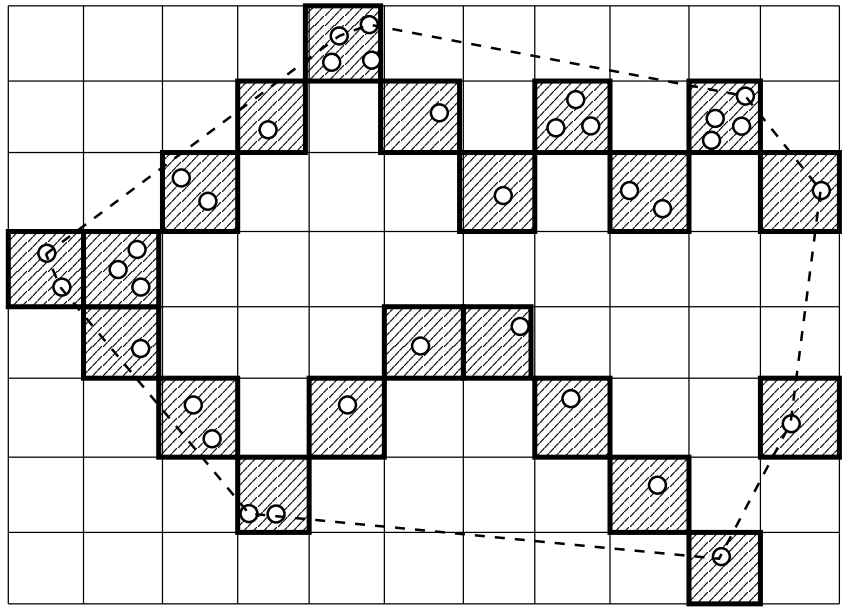
Algorithm 1: Grid

- Grid of cell size $\frac{\epsilon\alpha}{2\sqrt{d}}$



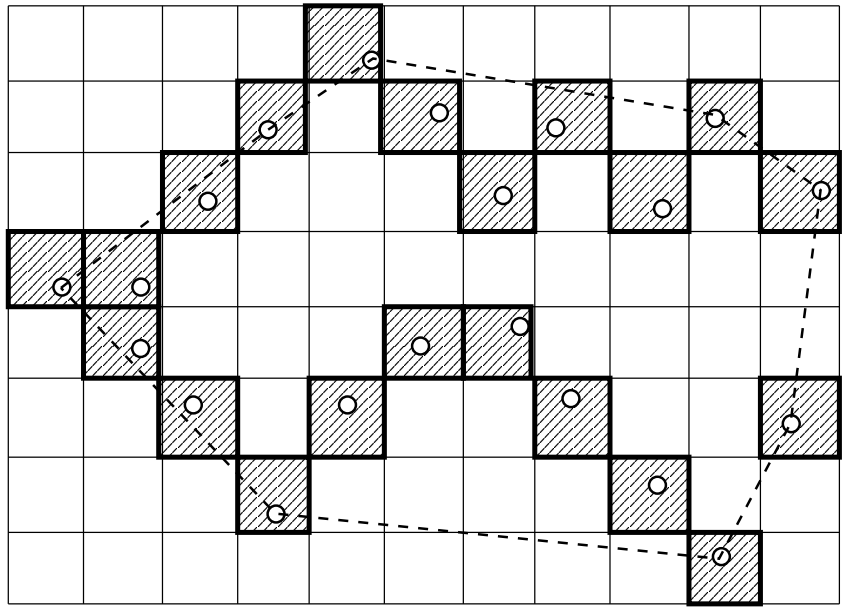
Algorithm 1: Grid

- Grid of cell size $\frac{\epsilon\alpha}{2\sqrt{d}}$
- Clear all "internal" cells
  - Convex hull moves by at most one cell diameter $\epsilon\alpha/2$



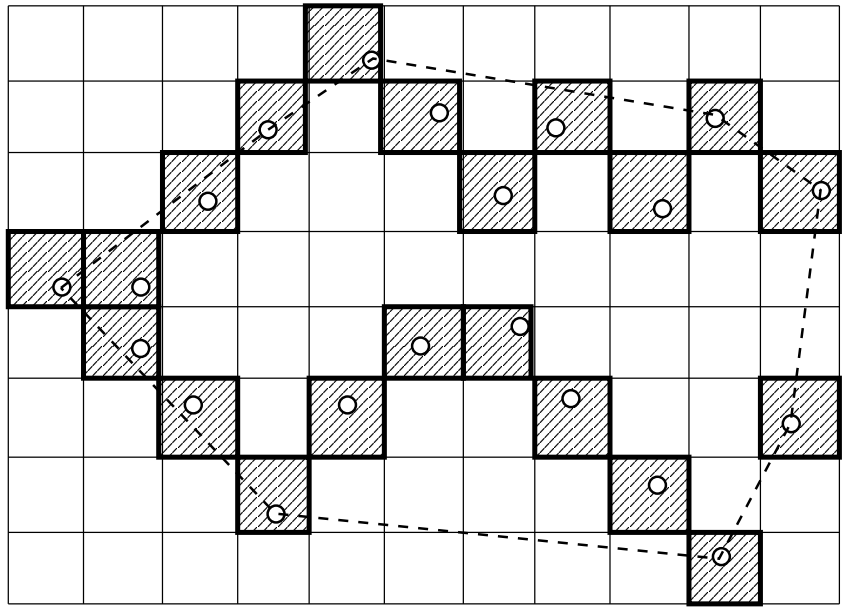
Algorithm 1: Grid

- Grid of cell size $\frac{\epsilon\alpha}{2\sqrt{d}}$
- Clear all "internal" cells
  - Convex hull moves by at most one cell diameter $\epsilon\alpha/2$
- Eliminate duplicates in the "boundary" cells
  - Another shift of at most $\epsilon\alpha/2$



Algorithm 1: Grid

- Grid of cell size $\frac{\epsilon\alpha}{2\sqrt{d}}$
- Clear all "internal" cells
  - Convex hull moves by at most one cell diameter $\epsilon\alpha/2$
- Eliminate duplicates in the "boundary" cells
  - Another shift of at most $\epsilon\alpha/2$



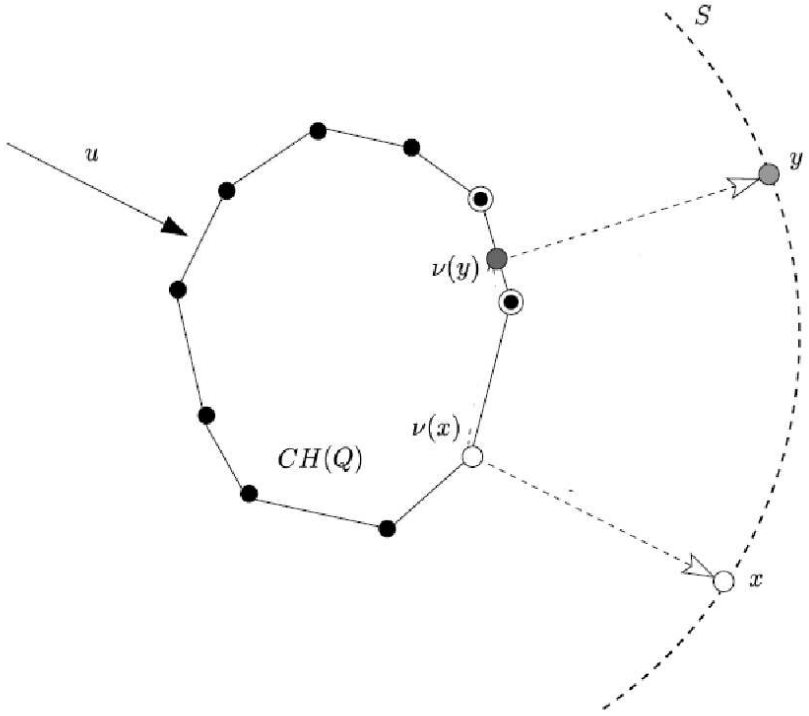
- Directional width changes at most $\epsilon\alpha$
  - Determined by convex hull

Algorithm 1: Analysis

- Coreset size $O(1/(\alpha\epsilon)^{d-1})$
  - One point per "boundary" grid cell
  - Cell size $O(1/(\alpha\epsilon))$
- Running time $O(n + 1/(\alpha\epsilon)^{d-1})$
- O-notation hides $\sqrt{d}$
- α has a bad dependence on d

Algorithm 2: Polytope

- Run Algorithm 1, return $(\epsilon/2)$-approximation P_1
- Apply [Dudley 1974] to P_1, lose another $\epsilon/2$
  - Sample the sphere of radius $\sqrt{d}+1$
  - Closest point routine [Gärtner 1995], runs in linear time, returns all (at most d) closest points
- Return the set of closest points P_2



Algorithm 2: Analysis

- Correctness
  - **Fact 1:** Dudley polytope is also valid for P_2
  - **Fact 2:** $\mathcal{CH}(P_2) \subseteq \mathcal{CH}(P_1) \subseteq \textsc{Dudley}$
- Want $(\alpha\epsilon/2)$-approximation of the convex hull
  - Yields $\epsilon/2$ for directional width
- Required number of samples

$$O((\sqrt{\alpha\epsilon/2})^{d-1}) = O((\alpha\epsilon)^{(d-1)/2})$$

- This is also (roughly) the size of the result P_2
  - At most d per sample
  - Compare to Algorithm 1: $O((\alpha\epsilon)^{d-1})$

Algorithm 2: Analysis

- Find closest points for $O((\alpha\epsilon)^{(d-1)/2})$ samples
- The closest point routine [Gärtner 1995]
  - Linear in the number of points
  - In this case, $|P_1| = O((\alpha\epsilon)^{d-1})$
  - Total, $O((\alpha\epsilon)^{3(d-1)/2})$
- Including the call to Algorithm 1, we get $O(n + (\alpha\epsilon)^{3(d-1)/2})$
  - Compare to Algorithm 1: $O(n + (\alpha\epsilon)^{d-1})$

Algorithm 3: Arrangement

- Run Algorithm 2, get an $(\epsilon/2)$-approximation P_2
- Decompose a set of directions $\mathbb{R}^{d-1}$ using an arrangement of $(d-2)$-flats

  $u, v \in \mathbb{R}^{d-1}$ in the same cell $\quad \Rightarrow \quad \left\| \frac{u}{||u||} - \frac{v}{||v||} \right\| \leq \frac{\alpha\epsilon}{4\sqrt{d}}$

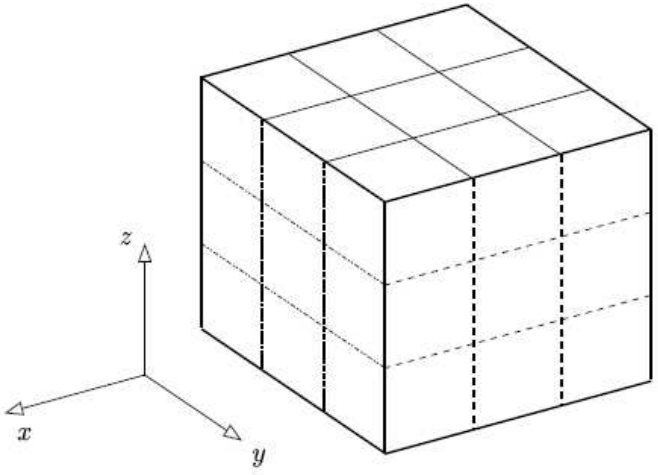
  - Identify diametrically opposite cells
    (induced partition of $\mathbb{P}^{d-1}$, the set of unoriented directions)

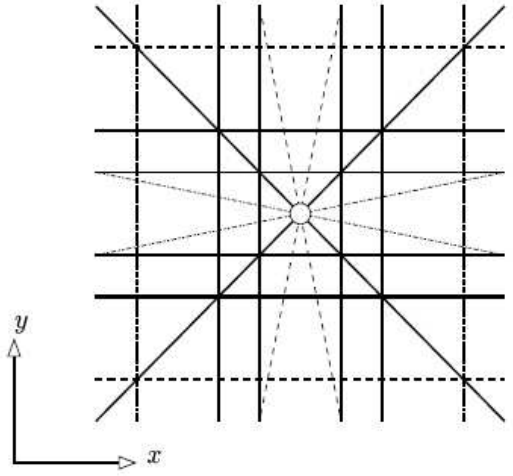
- Return P_3, consisting of two points per cell
  - Extremal points in one arbitrary direction within the cell
  - Lose another $\epsilon/2$

$$\langle \text{ANGLE ERROR}, \text{EXTREME POINT ERROR} \rangle \leq$$

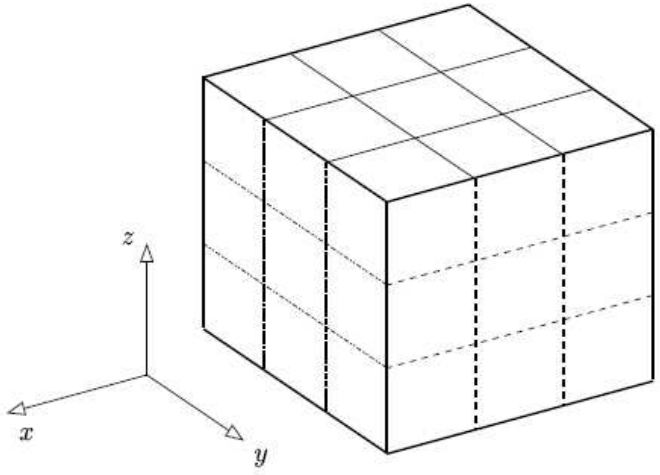
$$\leq \left\| \frac{u}{||u||} - \frac{v}{||v||} \right\| \cdot ||p - q|| \leq \frac{\alpha\epsilon}{4\sqrt{d}} \cdot 2\sqrt{d} = \frac{\alpha\epsilon}{2}$$

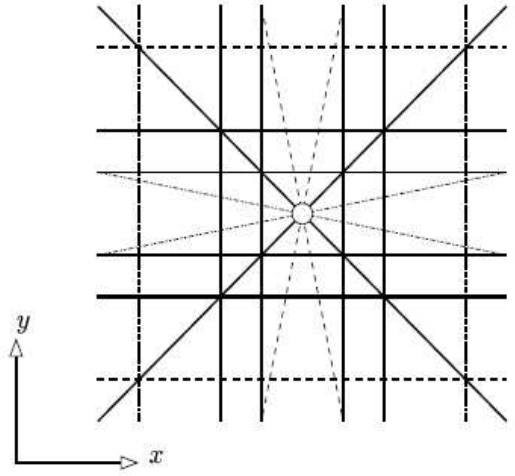
Algorithm 3: Computing the Arrangement





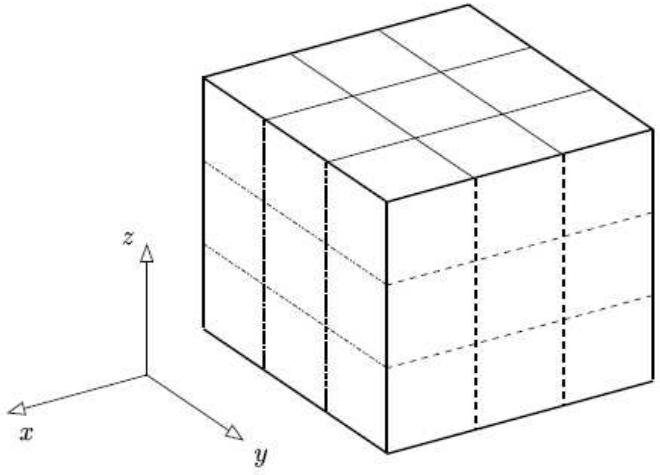
Algorithm 3: Computing the Arrangement

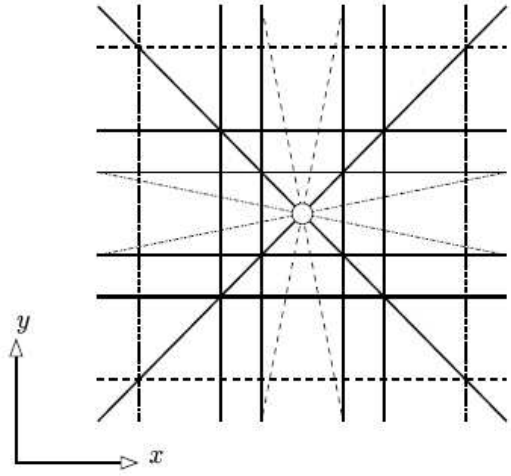




- $(d-1)$-dimensional grid within each of the d facets of $\mathbb{C}$
  - Cell size $\frac{\alpha\epsilon}{4\sqrt{d}}$
  - Results in a set of $(d-2)$-flats
- Affine hull with the origin, intersect with $\mathbb{P}^{d-1}$

Algorithm 3: Computing the Arrangement





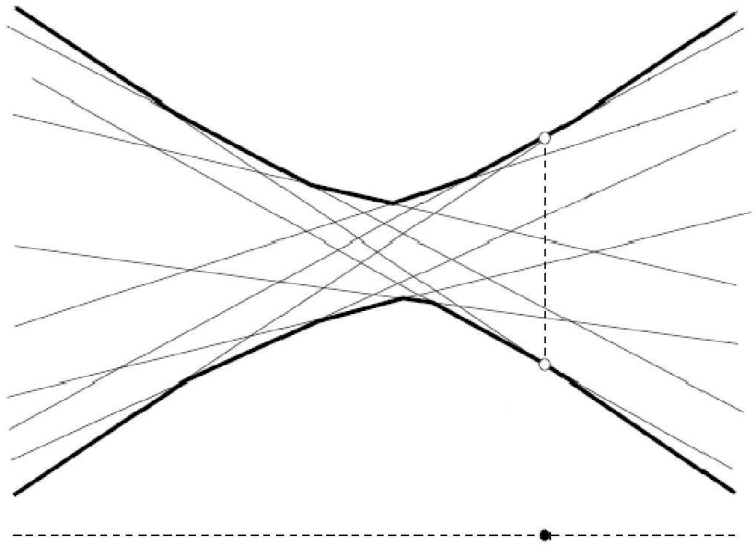
- $(d-1)$-dimensional grid within each of the d facets of $\mathbb{C}$
  - Cell size $\frac{\alpha\epsilon}{4\sqrt{d}}$
  - Results in a set of $(d-2)$-flats
- Affine hull with the origin, intersect with $\mathbb{P}^{d-1}$
- Complexity of the arrangement
  - Number of hyperplanes $d(d-1)\frac{4\sqrt{d}}{\alpha\epsilon}$
  - Complexity $O(1/(\alpha\epsilon)^{d-1})$

Algorithm 3: Analysis

- Coreset size $|P_3| = O(1/(\alpha\epsilon)^{d-1})$
  - Same as the arrangement complexity
- Extremal points are computed by linear search through P_2
  - $|P_2| = O(1/(\alpha\epsilon)^{(d-1)/2})$ time per cell
  - $O(1/(\alpha\epsilon)^{d-1})$ cells
  - $O(1/(\alpha\epsilon)^{3(d-1)/2})$ total
  - Combined with Algorithm 2, $O(n + 1/(\alpha\epsilon)^{3(d-1)/2})$
- Note $|P_3| > |P_2|$, a point may be extremal in multiple cells

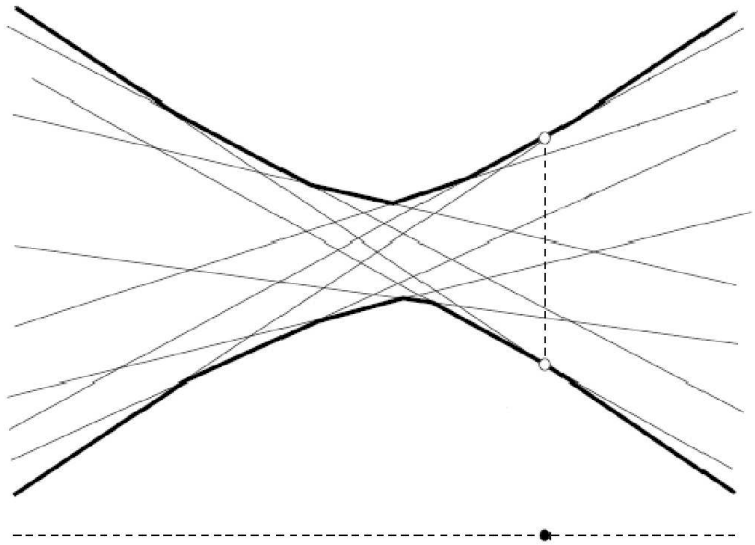
Approximating $(d-1)$-variate Functions

- Recall: ϵ-approximation in terms of extent



Approximating $(d-1)$-variate Functions

- Recall: ϵ-approximation in terms of extent



1. Linear functions (hyperplanes), using duality
2. General polynomials, using linearization
3. r-th roots of polynomials, $r \in \mathbb{Z}$

Point-Hyperplane Duality

- The hyperplane
  $$h: \; x_d = a_1 x_1 + a_2 x_2 + \cdots + a_{d-1} x_{d-1} + a_d$$
  corresponds to the point
  $$h^* = (a_1, a_2, \ldots, a_d) \in \mathbb{R}^d$$
- **Lemma:** Set of hyperplanes $H = \{h_1, h_2, \ldots, h_n\}$ is ϵ-approximated by $K \subset H$ over $\Delta \in \mathbb{R}^{d-1}$ iff H^* is ϵ-approximated by K^* over Δ
- **Proof:** Directional width $w(u, H^*)$ is the same as the extent $e(u, H)$
  - Holds for any $u \in \mathbb{R}^{d-1}$ and any set of hyperplanes H in $\mathbb{R}^d$
  - Important to define $w(\cdot, \cdot)$ using the $x_d = 1$ plane
- Immediately implies ϵ-approximation algorithms for linear functions in $(d-1)$-variables

Linearization

- A set of $(d-1)$-variate polynomials $F = \{f_1, f_2, \ldots, f_n\}$
- Linearization of dimension k

$$f_i(x) = a_0^{(i)} + a_1^{(i)}\phi_1(x) + \cdots + a_k^{(i)}\phi_k(x) \qquad i = 1, 2, \ldots, n$$

- Reduces to a set of k-variate linear functions

$$f_i(y) = a_0^{(i)} + a_1^{(i)}y_1 + \cdots + a_k^{(i)}y_k \qquad i = 1, 2, \ldots, n$$

  with $y = \phi(x)$, $\phi : \mathbb{R}^{d-1} \rightarrow \mathbb{R}^k$

- An ϵ-approximation for $\{f_i(y)\}$ over $\Delta \in \mathbb{R}^k$ implies an ϵ-approximation for $\{f_i(x)\}$ over $\phi^{-1}(\Delta \cap \phi(\mathbb{R}^{d-1}))$

Linearization Example

- $f_i(x)$ is the distance of the point $(x_1, x_2) \in \mathbb{R}^2$ to a circle in $\mathbb{R}^2$ with center $(p^{(i)}, q^{(i)})$ and radius $r^{(i)}$

$$f_i(x) = (r^{(i)})^2 - (x_1 - p^{(i)})^2 - (x_2 - q^{(i)})^2$$

- Can be written as

$$f_i(x) = (r^{(i)} - p^{(i)} - q^{(i)})^2 + (2p^{(i)})x_1 + (2q^{(i)})x_2 - [x_1^2 + x_2^2]$$

- A linearization of dimension $k = 3$ with

$$a^{(i)} = [(r^{(i)} - p^{(i)} - q^{(i)})^2,\ 2p^{(i)},\ 2q^{(i)},\ -1]$$

$$\phi(x) = [x_1,\ x_2,\ x_1^2 + x_2^2]$$

- [Agarwal, Matoušek 1994] Computing linearization of minimum dimension

Polynomials: Algorithms

- For a set F of n $(d-1)$-variate polynomials admits a linearization of dimension k, can compute
  - **Algorithm 1:** an ϵ-approximation of size $O(1/\epsilon^k)$, in time $O(n + 1/\epsilon^k)$
  - **Algorithm 2:** an ϵ-approximation of size $O(1/\epsilon^{k/2})$, in time $O(n + 1/\epsilon^{3k/2})$
  - **Algorithm 3:** a set of $O(1/\epsilon)(d-2)$-dimensional surfaces in $\mathbb{R}^{d-1}$, in time $O(n + 1/\epsilon^{3k/2})$, such that within each cell of their arrangement F is ϵ-approximated by 2 of its elements
  - Complexity of the arrangement $O(1/\epsilon^{d-1})$ [Agarwal, Sharir 00] Cells are of complexity $O(1)$
  - **Algorithms 2 and 3:** an ϵ-approximation of size $O(1/\epsilon^{\min\{k/2, d-1\}})$, in time $O(n + 1/\epsilon^{3k/2})$

Roots of Polynomials

- Want ϵ-approximation for $F = \{(f_1)^{1/r}, (f_2)^{1/r}, \ldots, (f_n)^{1/r}\}$ where r is integer
- Cannot linearize directly
- Special cases can be handled [Chan 02]
- If $a, b, A, B \geq 0$ and $[A, B] \subseteq [a, b]$ then

  $$B - A \geq (1-\delta)(b-a) \quad \Rightarrow \quad B^{1/r} - A^{1/r} \geq (1-\epsilon)(b^{1/r} - a^{1/r})$$

  for $\delta = \left(\frac{\epsilon}{2(r-1)}\right)^r$
- It suffices to compute $O(\epsilon^r)$ approximation to $\{f_1, f_2, \ldots, f_n\}$

Roots of Polynomials: Algorithms

- Set of F, $|F| = n$, contains r-th roots of $(d-1)$-variate non-negative polynomials that admit a linearization of of dimension k
- **Algorithm 1:** ϵ-approximation in time $O(1/\epsilon^{kr})$ and size $O(1/\epsilon^{kr})$
- **Algorithms 2 and 3:** ϵ-approximation in time $O(1/\epsilon^{3kr/2})$ and size $O(1/\epsilon^{r \min\{d-1,k/2\}})$
- **Algorithm 3:** A set of $O(1/\epsilon^{r})$ $(d-2)$-dimensional surfaces in $\mathbb{R}^{d-1}$, in time $O(1/\epsilon^{3kr/2})$, such that within each cell of their arrangement, F is ϵ-approximated by two of its elements

Classes of Extent Measures

- **Faithful**
  - Approximated through directional width
  - "Convex" measures (bounding shapes)
  - Examples:

    Diameter
    Width
    Radius of the smallest enclosing ball
    Volume of the minimum bounding box
    Volume enclosed by the convex hull
    Surface area of the convex hull

- **Other**
  - Approximated through extent
  - "Concave" measures ("shells")
  - Examples:

    Minimum width spherical shell (annulus)
    Minimum width cylindrical shell

Dynamic Updates

- Based on a **balanced tree** data structure
  - Both insertions and deletions
  - Recompute approximations along the unique path to the root
  - Update time per operation $O((\log^{k+1} n/\epsilon)^k + f(\log n/\epsilon)\log n)$
  - Amortized $O((\log^k n/\epsilon)^k + f(\log n/\epsilon))$
- Based on the **ranked subsets** data structure
  - Insertions only
  - Partition the points into ranked subsets, based on the binary encoding of n
  - Data structure size $O(\log^{2k+1} n/\epsilon^k)$
  - Coreset size $O(\log^{2k+1} n/\epsilon^k)$,
    amortized insertion time $O((1/\epsilon)^k + f(\epsilon))$
  - Coreset size $O(1/\epsilon^k)$,
    amortized insertion time $O(\log^{2k+1} n/\epsilon^k + f(\epsilon))$