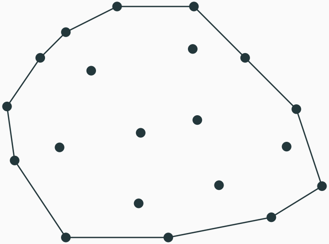
On the Budgeted Hausdorff Distance Problem

Sariel Har-Peled [1] *Benjamin Raichel* [2]

[1]UIUC

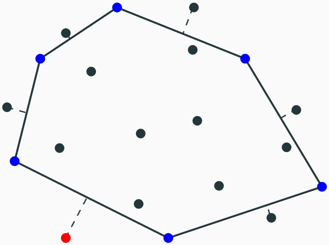
[2]UT Dallas

Convex Hull Approximation



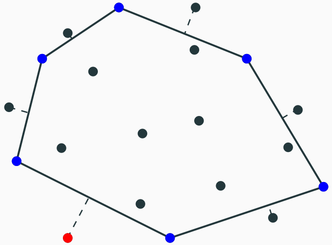
- Given a set P of n points in the plane, the convex hull $\mathcal{C}(P)$ is one the most fundamental objects in computational geometry.

Convex Hull Approximation



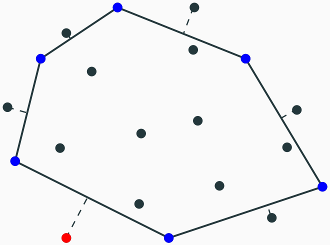
- Given a set P of n points in the plane, the convex hull $\mathcal{C}(P)$ is one the most fundamental objects in computational geometry.
- $\mathcal{C}(P)$ might contain many points, thus natural to consider finding a smaller subset $Q \subseteq P$ such that $\mathcal{C}(Q) \approx \mathcal{C}(P)$.

Convex Hull Approximation



- Given a set P of n points in the plane, the convex hull $\mathcal{C}(P)$ is one the most fundamental objects in computational geometry.
- $\mathcal{C}(P)$ might contain many points, thus natural to consider finding a smaller subset $Q \subseteq P$ such that $\mathcal{C}(Q) \approx \mathcal{C}(P)$.
- We will measure similarity using the Hausdorff distance between $\mathcal{C}(Q)$ and $\mathcal{C}(P)$, which we denote $\mathrm{D}_H(Q, P)$.

Convex Hull Approximation



- Given a set P of n points in the plane, the convex hull $\mathcal{C}(P)$ is one the most fundamental objects in computational geometry.
- $\mathcal{C}(P)$ might contain many points, thus natural to consider finding a smaller subset $Q \subseteq P$ such that $\mathcal{C}(Q) \approx \mathcal{C}(P)$.
- We will measure similarity using the Hausdorff distance between $\mathcal{C}(Q)$ and $\mathcal{C}(P)$, which we denote $\mathrm{D}_H(Q, P)$.
- Since $Q \subseteq P$, and $\mathcal{C}(Q), \mathcal{C}(P)$ convex, one can argue $\mathrm{D}_H(Q, P) = \max_{p \in P} ||p - \mathcal{C}(Q)||$.

Problem Variants

MinCardin

Given a set $P \subset \mathbb{R}^2$ of n points, and a value $\tau > 0$, find the smallest cardinality subset $Q \subseteq P$ such that $\mathrm{D}_H(Q, P) \leq \tau$.

$k^\star = k^\star(P, \tau) = \min_{Q \subseteq P : \mathrm{D}_H(Q,P) \leq \tau} |Q|$ is the min cardinality of cost τ.

Problem Variants

MinCardin

Given a set $P \subset \mathbb{R}^2$ of n points, and a value $\tau > 0$, find the smallest cardinality subset $Q \subseteq P$ such that $\mathrm{D}_H(Q, P) \leq \tau$.

$k^\star = k^\star(P, \tau) = \min_{Q \subseteq P : \mathrm{D}_H(Q,P) \leq \tau} |Q|$ is the min cardinality of cost τ.

MinDist

Given a set $P \subset \mathbb{R}^2$ of n points, and an integer k, find the subset $Q \subseteq P$ that minimizes $\mathrm{D}_H(Q, P)$ subject to $|Q| \leq k$.

$\tau^\star = \tau^\star(P, k) = \min_{Q \subseteq P : |Q| = k} \mathrm{D}_H(Q, P)$ is the optimal radius for k.

Problem Variants

MinCardin

Given a set $P \subset \mathbb{R}^2$ of n points, and a value $\tau > 0$, find the smallest cardinality subset $Q \subseteq P$ such that $\mathrm{D}_H(Q, P) \leq \tau$.

$k^\star = k^\star(P, \tau) = \min_{Q \subseteq P : \mathrm{D}_H(Q,P) \leq \tau} |Q|$ is the min cardinality of cost τ.

MinDist

Given a set $P \subset \mathbb{R}^2$ of n points, and an integer k, find the subset $Q \subseteq P$ that minimizes $\mathrm{D}_H(Q, P)$ subject to $|Q| \leq k$.

$\tau^\star = \tau^\star(P, k) = \min_{Q \subseteq P : |Q| = k} \mathrm{D}_H(Q, P)$ is the optimal radius for k.

MinCardin and MinDist are polynomial time solvable and are dual. A solution to one can be used to search for the solution to the other. Naively the solutions and searching are at least cubic time.

Prior Work

- [KR21] For points in convex position, $O(n \log^2(n))$ time for MinCardin and $O(cn \log^3(n))$ w.h.p. for MinDist. For arbitrary position, by reducing to unweighted APSP, $O(n^{2.5302})$ time for both.

Prior Work

- [KR21] For points in convex position, $O(n\log^2(n))$ time for MinCardin and $O(cn\log^3(n))$ w.h.p. for MinDist. For arbitrary position, by reducing to unweighted APSP, $O(n^{2.5302})$ time for both.
- Remark: Beating APSP time non-trivial. Must exploit geometry.

Prior Work

- [KR21] For points in convex position, $O(n\log^2(n))$ time for MinCardin and $O(cn\log^3(n))$ w.h.p. for MinDist. For arbitrary position, by reducing to unweighted APSP, $O(n^{2.5302})$ time for both.
- Remark: Beating APSP time non-trivial. Must exploit geometry.
- [AH23] Most relevant result, $O(k^\star n\log n)$ time for MinCardin.

Prior Work

- [KR21] For points in convex position, $O(n \log^2(n))$ time for MinCardin and $O(cn \log^3(n))$ w.h.p. for MinDist. For arbitrary position, by reducing to unweighted APSP, $O(n^{2.5302})$ time for both.
- Remark: Beating APSP time non-trivial. Must exploit geometry.
- [AH23] Most relevant result, $O(k^\star n \log n)$ time for MinCardin.
- Natural question: Can one get a similar improvement for MinDist.

Prior Work

- [KR21] For points in convex position, $O(n \log^2(n))$ time for MinCardin and $O(cn \log^3(n))$ w.h.p. for MinDist. For arbitrary position, by reducing to unweighted APSP, $O(n^{2.5302})$ time for both.
- Remark: Beating APSP time non-trivial. Must exploit geometry.
- [AH23] Most relevant result, $O(k^\star n \log n)$ time for MinCardin.
- Natural question: Can one get a similar improvement for MinDist.

Many other prior works on approximating the convex hull, though those works generally do not compute the optimal approximation, or give cubic time algorithms for Hausdorff or other related measures.

Decision Procedure

High level idea: solve MinDist, by searching using decider for MinCardin.

Decision Procedure

High level idea: solve MinDist, by searching using decider for MinCardin.

Theorem ([AH23])

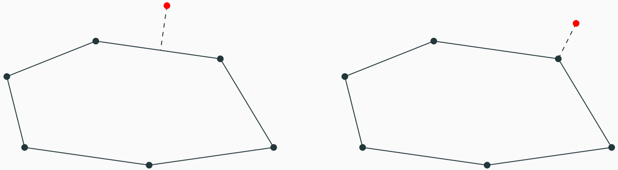
Given an instance P, k of MinDist and a value τ. Let $\tau^\star = \tau^\star(P, k)$.

There is a procedure **decider**(P, k, τ)*, that in $O(nk \log n)$ time returns*

1. $\tau = \tau^\star$,
2. $\tau < \tau^\star$, *or*
3. $\tau > \tau^\star$ *and a set Q where $|Q| \leq k$ and $\mathrm{D}_H(Q, P) \leq \tau$*

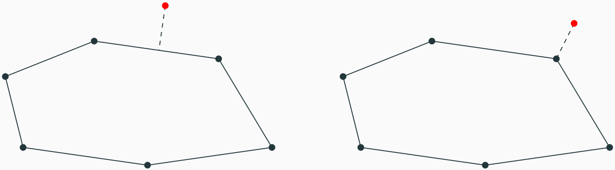
Note the result in [AH23] is actually for MinCardin, but with some massaging it yields the above decider.

Canonical Set of Values



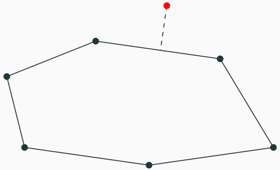
- Given an instance P, k of MinDist, the optimal value $\tau^\star$ is the distance of the point $p \in P$ furthest from $\mathcal{C}(Q^\star)$.

Canonical Set of Values



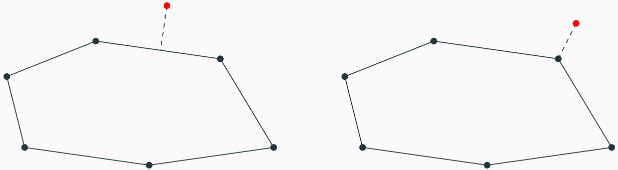
- Given an instance P, k of MinDist, the optimal value $\tau^\star$ is the distance of the point $p \in P$ furthest from $\mathcal{C}(Q^\star)$.
- p projects onto some edge ab of $\mathcal{C}(Q^\star)$, either at a, b, or the interior.

Canonical Set of Values



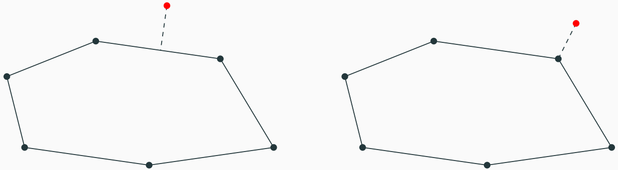
- Given an instance P, k of MinDist, the optimal value $\tau^\star$ is the distance of the point $p \in P$ furthest from $\mathcal{C}(Q^\star)$.
- p projects onto some edge ab of $\mathcal{C}(Q^\star)$, either at a, b, or the interior.
- Thus $\tau^\star$ is either the distance from p to another point in P, or the distance from p to the line $\ell_{a,b}$ for some pair $a, b \in P$.

Canonical Set of Values



- Given an instance P, k of MinDist, the optimal value $\tau^\star$ is the distance of the point $p \in P$ furthest from $\mathcal{C}(Q^\star)$.
- p projects onto some edge ab of $\mathcal{C}(Q^\star)$, either at a, b, or the interior.
- Thus $\tau^\star$ is either the distance from p to another point in P, or the distance from p to the line $\ell_{a,b}$ for some pair $a, b \in P$.
- Let $P_{a,b}$ be subset of P left of $\ell_{a,b}$. Can argue if $\tau^\star$ is the distance to $\ell_{a,b}$, then p is point from $P_{a,b}$ furthest from $\ell_{a,b}$.

Canonical Set of Values



- Given an instance P, k of MinDist, the optimal value $\tau^\star$ is the distance of the point $p \in P$ furthest from $\mathcal{C}(Q^\star)$.
- p projects onto some edge ab of $\mathcal{C}(Q^\star)$, either at a, b, or the interior.
- Thus $\tau^\star$ is either the distance from p to another point in P, or the distance from p to the line $\ell_{a,b}$ for some pair $a, b \in P$.
- Let $P_{a,b}$ be subset of P left of $\ell_{a,b}$. Can argue if $\tau^\star$ is the distance to $\ell_{a,b}$, then p is point from $P_{a,b}$ furthest from $\ell_{a,b}$.
- Thus $\tau^\star \in \mathcal{V} \cup \mathcal{L}$, where

$$\mathcal{V} = \left\{ ||x - y|| \;\middle|\; x, y \in P \right\} \quad \text{and} \quad \mathcal{L} = \left\{ \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b}) \;\middle|\; a, b \in P \right\}.$$

Searching

- Let **extremal**$(\ell_{a,b}) = \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**$(\ell_{a,b})$ takes $O(\log n)$ time by binary searching.

Searching

- Let **extremal**$(\ell_{a,b}) = \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**$(\ell_{a,b})$ takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.

Searching

- Let **extremal**$(\ell_{a,b}) = \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**$(\ell_{a,b})$ takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.
- Gives $O(n^2 \log n + nk \log^2 n)$ using the $O(nk \log n)$ time **decider**.

Searching

- Let **extremal**$(\ell_{a,b}) = \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**$(\ell_{a,b})$ takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.
- Gives $O(n^2 \log n + nk \log^2 n)$ using the $O(nk \log n)$ time **decider**.
- Our goal is to beat quadratic time for small k.

Searching

- Let **extremal**$(\ell_{a,b}) = \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**$(\ell_{a,b})$ takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.
- Gives $O(n^2 \log n + nk \log^2 n)$ using the $O(nk \log n)$ time **decider**.
- Our goal is to beat quadratic time for small k.

Theorem ([CZ21])

Given a set $P \subset \mathbb{R}^2$ of n points, and an integer $k > 0$, with high probability, in $O(n^{4/3})$ time, one can compute the value of rank k in $\mathcal{V}$.

Searching

- Let **extremal**$(\ell_{a,b}) = \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**$(\ell_{a,b})$ takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.
- Gives $O(n^2 \log n + nk \log^2 n)$ using the $O(nk \log n)$ time **decider**.
- Our goal is to beat quadratic time for small k.

Theorem ([CZ21])

Given a set $P \subset \mathbb{R}^2$ of n points, and an integer $k > 0$, with high probability, in $O(n^{4/3})$ time, one can compute the value of rank k in $\mathcal{V}$.

Stage I of the algorithm:

Searching

- Let **extremal**$(\ell_{a,b}) = \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**$(\ell_{a,b})$ takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.
- Gives $O(n^2 \log n + nk \log^2 n)$ using the $O(nk \log n)$ time **decider**.
- Our goal is to beat quadratic time for small k.

Theorem ([CZ21])

Given a set $P \subset \mathbb{R}^2$ of n points, and an integer $k > 0$, with high probability, in $O(n^{4/3})$ time, one can compute the value of rank k in $\mathcal{V}$.

Stage I of the algorithm:

- Use **decider** to binary search over $\mathcal{V}$ using [CZ21].

Searching

- Let **extremal**($\ell_{a,b}$) $= \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**($\ell_{a,b}$) takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.
- Gives $O(n^2 \log n + nk \log^2 n)$ using the $O(nk \log n)$ time **decider**.
- Our goal is to beat quadratic time for small k.

Theorem ([CZ21])

Given a set $P \subset \mathbb{R}^2$ of n points, and an integer $k > 0$, with high probability, in $O(n^{4/3})$ time, one can compute the value of rank k in $\mathcal{V}$.

Stage I of the algorithm:

- Use **decider** to binary search over $\mathcal{V}$ using [CZ21].
- If $\tau^\star \in \mathcal{V}$, then we will find $\tau^\star$ and terminate.

Searching

- Let **extremal**($\ell_{a,b}$) $= \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**($\ell_{a,b}$) takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.
- Gives $O(n^2 \log n + nk \log^2 n)$ using the $O(nk \log n)$ time **decider**.
- Our goal is to beat quadratic time for small k.

Theorem ([CZ21])

Given a set $P \subset \mathbb{R}^2$ of n points, and an integer $k > 0$, with high probability, in $O(n^{4/3})$ time, one can compute the value of rank k in $\mathcal{V}$.

Stage I of the algorithm:

- Use **decider** to binary search over $\mathcal{V}$ using [CZ21].
- If $\tau^\star \in \mathcal{V}$, then we will find $\tau^\star$ and terminate.
- Otherwise, returns value $r, R \in \mathcal{V}$ such that $\tau^\star \in (r, R)$.

Searching

- Let **extremal**$(\ell_{a,b}) = \max_{p \in P_{a,b}} \mathrm{d}(p, \ell_{a,b})$. If one precomputes $\mathcal{C}(P)$, **extremal**$(\ell_{a,b})$ takes $O(\log n)$ time by binary searching.
- $|\mathcal{V} \cup \mathcal{L}| = O(n^2)$, and takes $O(n^2 \log n)$ time to explicitly compute.
- Gives $O(n^2 \log n + nk \log^2 n)$ using the $O(nk \log n)$ time **decider**.
- Our goal is to beat quadratic time for small k.

Theorem ([CZ21])

Given a set $P \subset \mathbb{R}^2$ of n points, and an integer $k > 0$, with high probability, in $O(n^{4/3})$ time, one can compute the value of rank k in $\mathcal{V}$.

Stage I of the algorithm:

- Use **decider** to binary search over $\mathcal{V}$ using [CZ21].
- If $\tau^\star \in \mathcal{V}$, then we will find $\tau^\star$ and terminate.
- Otherwise, returns value $r, R \in \mathcal{V}$ such that $\tau^\star \in (r, R)$.
- Takes $O((n^{4/3} + nk \log n) \log n)$ time w.h.p.

Stage 2

- After Stage I, we can now focus on $\mathcal{L}$.

Stage 2

- After Stage I, we can now focus on $\mathcal{L}$.
- Sample a set S of $\Theta(n^{3/2} \log n)$ values from $\mathcal{L}$. Let $U = S \cap (r, R)$.

Stage 2

- After Stage I, we can now focus on $\mathcal{L}$.
- Sample a set S of $\Theta(n^{3/2} \log n)$ values from $\mathcal{L}$. Let $U = S \cap (r, R)$.
- Binary search over U, again using **decider**.

Stage 2

- After Stage I, we can now focus on $\mathcal{L}$.
- Sample a set S of $\Theta(n^{3/2} \log n)$ values from $\mathcal{L}$. Let $U = S \cap (r, R)$.
- Binary search over U, again using **decider**.
- Again produces an interval (r', R') such that $\tau^{\star} \in (r', R')$.

Stage 2

- After Stage I, we can now focus on $\mathcal{L}$.
- Sample a set S of $\Theta(n^{3/2} \log n)$ values from $\mathcal{L}$. Let $U = S \cap (r, R)$.
- Binary search over U, again using **decider**.
- Again produces an interval (r', R') such that $\tau^\star \in (r', R')$.
- Can argue that w.h.p. $(r', R') \cap \mathcal{L} = O(\sqrt{n})$. (similar to [HR14]).

Stage 2

- After Stage I, we can now focus on $\mathcal{L}$.
- Sample a set S of $\Theta(n^{3/2} \log n)$ values from $\mathcal{L}$. Let $U = S \cap (r, R)$.
- Binary search over U, again using **decider**.
- Again produces an interval (r', R') such that $\tau^\star \in (r', R')$.
- Can argue that w.h.p. $(r', R') \cap \mathcal{L} = O(\sqrt{n})$. (similar to [HR14]).
- The problem is we do not have direct access to the set $(r', R') \cap \mathcal{L}$.

Stage 3

- We know $\tau^\star \in (r', R')$. Since $R' > \tau^\star$, **decider**(P, k, R') returns a set Q where $|Q| \leq k$ and $\mathrm{D}_H(Q, P) \leq R'$.

Stage 3

- We know $\tau^\star \in (r', R')$. Since $R' > \tau^\star$, **decider**(P, k, R') returns a set Q where $|Q| \leq k$ and $\mathsf{D}_H(Q, P) \leq R'$.
- Let $\alpha = \mathsf{D}_H(Q, P)$. If $\alpha < R'$, check if $\tau^\star = \alpha$, and if not then $\tau^\star \in (r', \alpha)$ and we restart stage 3 with this interval.

Stage 3

- We know $\tau^\star \in (r', R')$. Since $R' > \tau^\star$, **decider**(P, k, R') returns a set Q where $|Q| \leq k$ and $\mathsf{D}_H(Q, P) \leq R'$.
- Let $\alpha = \mathsf{D}_H(Q, P)$. If $\alpha < R'$, check if $\tau^\star = \alpha$, and if not then $\tau^\star \in (r', \alpha)$ and we restart stage 3 with this interval.
- Assume $\alpha = R'$. We know $\tau^\star < \alpha$, thus $\tau^\star < \alpha - \varepsilon$ for an infinitesimal ε. Thus **decider**$(P, k, \alpha - \varepsilon)$ returns a set Q' such that $|Q'| \leq k$ and $\beta = \mathsf{D}_H(Q', P) < \alpha$.

Stage 3

- We know $\tau^\star \in (r', R')$. Since $R' > \tau^\star$, **decider**(P, k, R') returns a set Q where $|Q| \leq k$ and $\mathrm{D}_H(Q, P) \leq R'$.
- Let $\alpha = \mathrm{D}_H(Q, P)$. If $\alpha < R'$, check if $\tau^\star = \alpha$, and if not then $\tau^\star \in (r', \alpha)$ and we restart stage 3 with this interval.
- Assume $\alpha = R'$. We know $\tau^\star < \alpha$, thus $\tau^\star < \alpha - \varepsilon$ for an infinitesimal ε. Thus **decider**$(P, k, \alpha - \varepsilon)$ returns a set Q' such that $|Q'| \leq k$ and $\beta = \mathrm{D}_H(Q', P) < \alpha$.
- Check if $\beta = \tau^\star$, and if not then $\tau^\star \in (r', \beta)$ and so repeat stage 3 on the interval.

Stage 3

- We know $\tau^\star \in (r', R')$. Since $R' > \tau^\star$, **decider**(P, k, R') returns a set Q where $|Q| \leq k$ and $\mathrm{D}_H(Q, P) \leq R'$.
- Let $\alpha = \mathrm{D}_H(Q, P)$. If $\alpha < R'$, check if $\tau^\star = \alpha$, and if not then $\tau^\star \in (r', \alpha)$ and we restart stage 3 with this interval.
- Assume $\alpha = R'$. We know $\tau^\star < \alpha$, thus $\tau^\star < \alpha - \varepsilon$ for an infinitesimal ε. Thus **decider**$(P, k, \alpha - \varepsilon)$ returns a set Q' such that $|Q'| \leq k$ and $\beta = \mathrm{D}_H(Q', P) < \alpha$.
- Check if $\beta = \tau^\star$, and if not then $\tau^\star \in (r', \beta)$ and so repeat stage 3 on the interval.
- This open interval doesn't contain β and so has fewer values from $\mathcal{L}$.

Stage 3

- We know $\tau^\star \in (r', R')$. Since $R' > \tau^\star$, **decider**(P, k, R') returns a set Q where $|Q| \le k$ and $\mathrm{D}_H(Q, P) \le R'$.
- Let $\alpha = \mathrm{D}_H(Q, P)$. If $\alpha < R'$, check if $\tau^\star = \alpha$, and if not then $\tau^\star \in (r', \alpha)$ and we restart stage 3 with this interval.
- Assume $\alpha = R'$. We know $\tau^\star < \alpha$, thus $\tau^\star < \alpha - \varepsilon$ for an infinitesimal ε. Thus **decider**$(P, k, \alpha - \varepsilon)$ returns a set Q' such that $|Q'| \le k$ and $\beta = \mathrm{D}_H(Q', P) < \alpha$.
- Check if $\beta = \tau^\star$, and if not then $\tau^\star \in (r', \beta)$ and so repeat stage 3 on the interval.
- This open interval doesn't contain β and so has fewer values from $\mathcal{L}$.
- Thus in total Stage 3 runs for $O(\sqrt{n})$ rounds, each of which costs $O(nk \log n)$ time as this is the time for **decider**

Total Time

- Stage I: $O((n^{4/3} + nk \log n) \log n)$

Total Time

- Stage I: $O((n^{4/3} + nk \log n) \log n)$
- Stage II: $O((n^{3/2} \log n + nk \log n) \log n)$

Total Time

- Stage I: $O((n^{4/3} + nk \log n) \log n)$
- Stage II: $O((n^{3/2} \log n + nk \log n) \log n)$
- Stage III: $O(n^{3/2} k \log n)$

Total Time

- Stage I: $O((n^{4/3} + nk \log n) \log n)$
- Stage II: $O((n^{3/2} \log n + nk \log n) \log n)$
- Stage III: $O(n^{3/2} k \log n)$
- Total Time: $O(n^{3/2}(k + \log n) \log n)$

Total Time

- Stage I: $O((n^{4/3} + nk \log n) \log n)$
- Stage II: $O((n^{3/2} \log n + nk \log n) \log n)$
- Stage III: $O(n^{3/2} k \log n)$
- Total Time: $O(n^{3/2}(k + \log n) \log n)$
- Optimizing the $\mathcal{L}$ sample size can improve the time to: $O\big(n^{3/2}\sqrt{k} \log^{3/2} n + kn \log^2 n\big)$

Total Time

- Stage I: $O((n^{4/3} + nk \log n) \log n)$
- Stage II: $O((n^{3/2} \log n + nk \log n) \log n)$
- Stage III: $O(n^{3/2} k \log n)$
- Total Time: $O(n^{3/2}(k + \log n) \log n)$
- Optimizing the $\mathcal{L}$ sample size can improve the time to: $O\big(n^{3/2}\sqrt{k} \log^{3/2} n + kn \log^2 n\big)$

Main Result

The MinDist problem can be solved in $O\big(n^{3/2}\sqrt{k} \log^{3/2} n + kn \log^2 n\big)$ time with high probability.

Thank You

Pankaj K. Agarwal and Sariel Har-Peled.
Computing optimal kernels in two dimensions.
In *Proc. 39th Int. Annu. Sympos. Comput. Geom.* (SoCG), LIPIcs, page to appear. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

Timothy M. Chan and Da Wei Zheng.
Hopcroft's problem, log-star shaving, 2d fractional cascading, and decision trees.
CoRR, abs/2111.03744, 2021.

Sariel Har-Peled and Benjamin Raichel.
The Fréchet distance revisited and extended.
ACM Trans. Algorithms, 10(1):3:1–3:22, 2014.

Georgiy Klimenko and Benjamin Raichel.
Fast and exact convex hull simplification.
In Mikolaj Bojanczyk and Chandra Chekuri, editors, *Proc. 41th Conf. Found. Soft. Tech. Theoret. Comput. Sci.* (FSTTCS), volume 213

of *LIPIcs*, pages 26:1–26:17, Wadern, Germany, 2021. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.