Journey to the Center of the Point Set

Sariel Har-Peled
Dept. of Computer Science, University of Illinois at Urbana-Champaign, Urbana, USA
https://sarielhp.org/
sariel@illinois.edu

Mitchell Jones
Dept. of Computer Science, University of Illinois at Urbana-Champaign, Urbana, USA
http://mfjones2.web.engr.illinois.edu/
mfjones2@illinois.edu

Abstract

We revisit an algorithm of Clarkson et al. [1], that computes (roughly) a $1/(4d^2)$-centerpoint in $\widetilde{O}(d^9)$ time, for a point set in $\mathbb{R}^d$, where $\widetilde{O}$ hides polylogarithmic terms. We present an improved algorithm that computes (roughly) a $1/d^2$-centerpoint with running time $\widetilde{O}(d^7)$. While the improvements are (arguably) mild, it is the first progress on this well known problem in over twenty years. The new algorithm is simpler, and the running time bound follows by a simple random walk argument, which we believe to be of independent interest. We also present several new applications of the improved centerpoint algorithm.

2012 ACM Subject Classification Theory of computation → Computational geometry; Theory of computation → Random walks and Markov chains

Keywords and phrases Computational geometry, Centerpoints, Random walks

Digital Object Identifier 10.4230/LIPIcs.SoCG.2019.41

Related Version A full version of the paper is available at https://arxiv.org/abs/1712.02949.

Funding *Sariel Har-Peled*: Supported in part by NSF AF awards CCF-1421231 and CCF-1217462.

1 Introduction

Notation

In the following $O(\cdot)$ hides constants that do not depend on the dimension. $O_d(\cdot)$ hides constants that depend on the dimension (usually badly – exponential or doubly exponential, or even worse). The notation $\widetilde{O}(\cdot)$ hides polylogarithmic factors, where the power of the polylog is independent of the dimension.

Computing centerpoints

A classical implication of Helly's theorem, is that for any set P of n points in $\mathbb{R}^d$, there is a $1/(d+1)$-centerpoint. Specifically, given a constant $\alpha \in (0,1)$, a point $\overline{c} \in \mathbb{R}^d$ is an α-centerpoint if all closed halfspaces containing $\overline{c}$ also contain at least αn points of P. It is currently unknown if one can compute a $\Omega(1/d)$-centerpoint in polynomial time (in the dimension). A randomized polynomial time algorithm was presented by Clarkson et al. [1], that computes (roughly) a $1/(4d^2)$-centerpoint in $\widetilde{O}(d^9)$ time.

Weak ε-nets

Consider the range space $(P, \mathcal{C})$, where P is a set of n points in $\mathbb{R}^d$, and $\mathcal{C}$ is the set of all convex shapes in $\mathbb{R}^d$. This range space has infinite VC dimension, and as such it is impervious to the standard ε-net constructions. *Weak ε-nets* bypass this issue by using points outside the point set. While there is significant amount of work on weak ε-nets, the constructions known are not easy and result in somewhat large sets. The state of the art is the work by Matoušek and Wagner [10], which shows a weak ε-net construction of size $O(\varepsilon^{-d}(\log \varepsilon^{-1})^{O(d^2 \log d)})$. Rubin recently improved this result for points in $\mathbb{R}^2$, proving the existence of weak ε-nets of size $O(\varepsilon^{-(1.5+\gamma)})$ for arbitrarily small $\gamma > 0$ [15]. Such a weak ε-net $\mathcal{W}$ has the guarantee that any convex set C that contains at least εn points of P, must contain at least one point of $\mathcal{W}$. See [13] for a recent survey of ε-nets and related concepts. See also the recent work by Rok and Smorodinsky [14] and references therein.

Basis of weak ε-nets

Mustafa and Ray [12] showed that one can pick a random sample S of size $c_d \varepsilon^{-1} \log \varepsilon^{-1}$ from P, and then compute a weak ε-net for P directly from S, showing that the size of the support needed to compute a weak ε-net is (roughly) the size of a regular ε-net. Unfortunately, the constant in their sample $c_d = O\big(d^d (\log d)^{cd^3 \log d}\big)$ is doubly exponential in the dimension. This constant c_d is related to the $((d+1)^2, d+1)$-Hadwiger-Debrunner number (the best known upper bounds on (p,q)-Hadwiger-Debrunner numbers can be found in [7, 6]).

In particular, all current results about weak ε-nets suffer from the "curse of dimensionality" and have constants that are at least doubly exponential in the dimension.

Our results

Let P be a set of n points in $\mathbb{R}^d$. In addition to the improved algorithm for computing approximate centerpoints, we also suggest two alternatives to weak ε-nets as applications, and obtain some related results:

(a) **Approximating centerpoints.** We revisit the algorithm of Clarkson et al. [1] for approximating a centerpoint. We present an improved algorithm, which is a variant of their algorithm which runs in $\widetilde{O}(d^7)$ time, and computes roughly a $1/(d+2)^2$-centerpoint. This improves both the running time, and the quality of centerpoint computed. While the improvements are small (a factor of d^2 roughly in the running time, and a factor of four in the centerpoint quality), we believe that the new algorithm is simpler. The analysis is cleaner, and is of independent interest. In particular, the analysis uses a random walk argument, which is quite uncommon in computational geometry, and (we believe) is of independent interest. See Theorem 18. This is the first improvement of the randomized algorithm of Clarkson et al. [1] in over twenty years. Miller and Sheehy also derandomized the algorithm of Clarkson et al., computing a $\Omega(1/d^2)$-centerpoint in time $n^{O(\log d)}$ [11].

(b) **Lowerbounding convex functions.** Given a convex function f in $\mathbb{R}^d$, such that one can compute its value and gradient at a point efficiently, we present an algorithm that computes quickly a realizable lower-bound on the value of f over P. Formally, the algorithm computes a point $q \in \mathbb{R}^d$, such that $f(q) \leq \min_{p \in P} f(p)$. The algorithm is somewhat similar in spirit to the ellipsoid algorithm. The running time of the algorithm is $\widetilde{O}\big(d^9\big)$. See Theorem 22.

(c) **Functional nets.** Let $\mathsf{C} \subseteq \mathbb{R}^d$ be a convex body. Suppose we are only given access to C via a separation oracle: given a query point q, the oracle either returns that q is in C, or alternatively, the oracle returns a hyperplane separating q and C. We show that a random sample of size

$$O\big(\varepsilon^{-1} d^3 \log d \log^3 \varepsilon^{-1} + \varepsilon^{-1} \log \varphi^{-1}\big) = \widetilde{O}\big(d^3/\varepsilon\big),$$

with probability $\geq 1 - \varphi$, can be used to decide if a query convex body C is ε-light. Formally, the algorithm, using only the sample, performs $O(d^2 \log \varepsilon^{-1})$ oracle queries – if any of the query points generated stabs C, then C is considered as (potentially) containing more than εn points. Alternatively, if all the queries missed C, then C contains less than εn points of P. The query points can be computed in polynomial time, and we emphasize that the dependency in the running time and sample size are polynomial in ε and d. See Theorem 28. As such, this result can be viewed as slightly mitigating the curse of dimensionality in the context of weak ε-nets.

(d) **Center nets.** Using the above, one can also construct a weak ε-net directly from such a sample – this improves over the result of Mustafa and Ray [12] as far as the dependency on the dimension is concerned. This is construction is described in Lemma 33.
Surprisingly, by using ideas from Theorem 28 one can get a stronger form of a weak ε-net, which we refer to as an (ε, α)-*center net*. Here $\alpha = \Omega(1/(d \log \varepsilon^{-1}))$ and one can compute a set $\mathcal{W}$ of size (roughly) $\widetilde{O}_d\big(\varepsilon^{-O(d^2)}\big)$, such that if a convex body C contains $\geq \varepsilon n$ points of P, then $\mathcal{W}$ contains a point q which is an α-centerpoint of $\mathsf{C} \cap P$. Namely, the net contains a point that stabs C in the "middle" as far as the point set $\mathsf{C} \cap P$. See Theorem 38.

Paper organization

The improved centerpoint approximation algorithm is described in Section 3. Two applications of the improved centerpoint algorithm are presented in Section 4. The construction of center nets is described in Section 5. Background and standard tools used are described in Section 2.

2 Background

2.1 Ranges spaces, VC dimension, samples and nets

The following is a quick survey of (standard) known results about ε-nets, ε-samples, and relative approximations [2].

▶ **Definition 1.** *A* ***range space*** S *is a pair* $(\widehat{\mathsf{X}}, \mathcal{R})$*, where* $\widehat{\mathsf{X}}$ *is a* ***ground set*** *(finite or infinite) and* $\mathcal{R}$ *is a (finite or infinite) family of subsets of* $\widehat{\mathsf{X}}$*. The elements of* $\widehat{\mathsf{X}}$ *are* ***points*** *and the elements of* $\mathcal{R}$ *are* ***ranges****.*

For technical reasons, it will be easier to consider a finite subset $\mathsf{X} \subseteq \widehat{\mathsf{X}}$ as the underlining ground set.

▶ **Definition 2.** *Let* $\mathsf{S} = (\widehat{\mathsf{X}}, \mathcal{R})$ *be a range space, and let* X *be a finite (fixed) subset of* $\widehat{\mathsf{X}}$*. For a range* $\mathsf{r} \in \mathcal{R}$*, its* ***measure*** *is the quantity* $\overline{m}(\mathsf{r}) = |\mathsf{r} \cap \mathsf{X}| / |\mathsf{X}|$*. For a subset* $S \subseteq \mathsf{X}$*, its* ***estimate*** *of* $\overline{m}(\mathsf{r})$*, for* $\mathsf{r} \in \mathcal{R}$*, is the quantity* $\overline{s}(\mathsf{r}) = |\mathsf{r} \cap S| / |S|$*.*

▶ **Definition 3.** *Let* $\mathsf{S} = (\widehat{\mathsf{X}}, \mathcal{R})$ *be a range space. For* $Y \subseteq \widehat{\mathsf{X}}$*, let* $\mathcal{R}_{|Y} = \left\{ \mathsf{r} \cap Y \;\middle|\; \mathsf{r} \in \mathcal{R} \right\}$ *denote the* ***projection*** *of* $\mathcal{R}$ *on* Y*. The range space* S *projected to* Y *is* $\mathsf{S}_{|Y} = \big(Y, \mathcal{R}_{|Y}\big)$*. If* $\mathcal{R}_{|Y}$ *contains all subsets of* Y *(i.e., if* Y *is finite, we have* $\left|\mathcal{R}_{|Y}\right| = 2^{|Y|}$*), then* Y *is* ***shattered*** *by* $\mathcal{R}$ *(or equivalently* Y *is shattered by* S*).*

The ***VC dimension*** *of* S*, denoted by* $\dim_{\mathsf{VC}}(\mathsf{S})$*, is the maximum cardinality of a shattered subset of* $\widehat{\mathsf{X}}$*. If there are arbitrarily large shattered subsets, then* $\dim_{\mathsf{VC}}(S) = \infty$*.*

▶ **Definition 4.** *Let* $\mathsf{S} = (\widehat{\mathsf{X}}, \mathcal{R})$ *be a range space, and let* X *be a finite subset of* $\widehat{\mathsf{X}}$*. For* $0 \leq \varepsilon \leq 1$*, a subset* $S \subseteq \mathsf{X}$ *is an* ***ε-sample*** *for* X *if for any range* $\mathsf{r} \in \mathcal{R}$*, we have* $|\overline{m}(\mathsf{r}) - \overline{s}(\mathsf{r})| \leq \varepsilon$*, see Definition 2. Similarly, a set* $S \subseteq \mathsf{X}$ *is an* ***ε-net*** *for* X *if for any range* $\mathsf{r} \in \mathcal{R}$*, if* $\overline{m}(\mathsf{r}) \geq \varepsilon$ *(i.e.,* $|\mathsf{r} \cap \mathsf{X}| \geq \varepsilon |\mathsf{X}|$*), then* r *contains at least one point of* S *(i.e.,* $\mathsf{r} \cap S \neq \emptyset$*).*

A generalization of both concepts is relative approximation. *Let $p, \widehat{\varepsilon} > 0$ be two fixed constants. A* ***relative*** $(p, \widehat{\varepsilon})$***-approximation*** *is a subset $S \subseteq \mathsf{X}$ that satisfies $(1-\widehat{\varepsilon})\overline{m}(\mathsf{r}) \leq \overline{s}(\mathsf{r}) \leq (1+\widehat{\varepsilon})\overline{m}(\mathsf{r})$, for any $\mathsf{r} \in \mathcal{C}$ such that $\overline{m}(\mathsf{r}) \geq p$. If $\overline{m}(\mathsf{r}) < p$ then the requirement is that $|\overline{s}(\mathsf{r}) - \overline{m}(\mathsf{r})| \leq \widehat{\varepsilon} p$.*

▶ **Theorem 5** (ε-net theorem, [5]). *Let $(\widehat{\mathsf{X}}, \mathcal{R})$ be a range space of VC dimension ξ, let X be a finite subset of $\widehat{\mathsf{X}}$, and suppose that $0 < \varepsilon \leq 1$ and $\varphi < 1$. Let N be a set obtained by m random independent draws from X, where $m \geq \max\left(\frac{4}{\varepsilon}\lg\frac{4}{\varphi}, \frac{8\xi}{\varepsilon}\lg\frac{16}{\varepsilon}\right)$. Then N is an ε-net for X with probability at least $1-\varphi$.*

The following is a slight strengthening of the result of Vapnik and Chervonenkis [16] – see [2, Theorem 7.13].

▶ **Theorem 6** (ε-sample theorem). *Let $\varphi, \varepsilon > 0$ be parameters and let $(\widehat{\mathsf{X}}, \mathcal{R})$ be a range space with VC dimension ξ. Let $\mathsf{X} \subset \widehat{\mathsf{X}}$ be a finite subset. A sample of size $O\big(\varepsilon^{-2}(\xi + \log\varphi^{-1})\big)$ from X is an ε-sample for $\mathsf{S} = (\mathsf{X}, \mathcal{R})$ with probability $\geq 1-\varphi$.*

▶ **Theorem 7** ([8, 3]). *A sample S of size $O\big(\widehat{\varepsilon}^{-2}p^{-1}(\xi\log p^{-1} + \log\varphi^{-1})\big)$ from a range space with VC dimension ξ, is a relative $(p, \widehat{\varepsilon})$-approximation with probability $\geq 1-\varphi$.*

The following is a standard statement on the VC dimension of a range space formed by mixing several range spaces together (see [2]).

▶ **Lemma 8.** *Let $\mathsf{S}_1 = (\widehat{\mathsf{X}}_1, \mathcal{C}_1), \ldots, \mathsf{S}_k = (\widehat{\mathsf{X}}, \mathcal{C}_k)$ be k range spaces, which all have VC dimension ξ. Consider the new set of ranges $\widehat{\mathcal{C}} = \{r_1 \cap \ldots \cap r_k \mid r_1 \in R_1, \ldots, r_k \in R_k\}$. Then the range space $\widehat{\mathsf{S}} = (\widehat{\mathsf{X}}, \widehat{\mathcal{C}})$ has VC dimension $O(\xi k \log k)$.*

2.2 Weak ε-nets

A convex body $\mathsf{C} \subseteq \mathbb{R}^d$ is $\boldsymbol{\varepsilon}$**-heavy** (or just *heavy*) if $\overline{m}(\mathsf{C}) \geq \varepsilon$ (i.e., $|\mathsf{C} \cap P| \geq \varepsilon|P|$). Otherwise, C is $\boldsymbol{\varepsilon}$**-light**.

▶ **Definition 9** (Weak ε-net). *Let P be a set of n points in $\mathbb{R}^d$. A finite set $S \subset \mathbb{R}^d$ is a* ***weak*** $\boldsymbol{\varepsilon}$***-net*** *for P if for any convex set C with $\overline{m}(\mathsf{C}) \geq \varepsilon$, we have $S \cap \mathsf{C} \neq \varnothing$.*

Note, that like (regular) ε-nets, weak ε-nets have one-sided error – if C is heavy then the net must stab it, but if C is light then the net may or may not stab it.

2.3 Centerpoints

Given a set P of n points in $\mathbb{R}^d$, and a constant $\alpha \in (0, 1/(d+1)]$, a point $\overline{c} \in \mathbb{R}^d$ is an $\boldsymbol{\alpha}$**-centerpoint** if for any closed halfspace that contains $\overline{c}$, the halfspace also contains at least αn points of P. It is a classical consequence of Helly's theorem that a $1/(d+1)$-centerpoint always exists. If a point $\overline{c} \in \mathbb{R}^d$ is a $1/(d+1)$-centerpoint for P, we omit the $1/(d+1)$ and simply say that $\overline{c}$ is a centerpoint for P.

3 Approximating the centerpoint via Radon's urn

We revisit the algorithm of Clarkson et al. [1] for approximating a centerpoint. We give a variant of their algorithm, and present a different (and we believe cleaner) analysis of the algorithm. In the process we improve the running time from being roughly $\widetilde{O}(d^9)$ to $\widetilde{O}(d^7)$, and also improve the quality of centerpoint computed.

3.1 Radon's urn

3.1.1 Setup

In the **Radon's urn** game there are r red balls, and $b = n - r$ blue balls in an urn, and there is a parameter t. An iteration of the game goes as follows:

(a) The player picks a random ball, marks it for deletion, and returns it to the urn.
(b) The player picks a sample S of t balls (with replacement – which implies that we might have several copies of the same ball in the sample) from the urn.
(c) If at least two of the balls in the sample S are red, the player inserts a new red ball into the urn. Otherwise, the player inserts a new blue ball.
(d) The player returns the sample to the urn.
(e) Finally, the player removes the ball marked for deletion from the urn.

Note that in each stage of the game, the number of balls in the urn remains the same. We are interested in how many rounds of the game one has to play till there are no red balls in the urn with high probability. Here, the initial value of r (i.e., r_0) is going to be relatively small compared to n.

3.2 Analysis

Let $P(r)$ be the probability of picking two or more red balls into the sample, assuming that there are r red balls in the urn. We have that

$$P(r) = \sum_{i=2}^{t} \binom{t}{i} \left(\frac{r}{n}\right)^i \left(1 - \frac{r}{n}\right)^{t-i} \leq \binom{t}{2} \left(\frac{r}{n}\right)^2 \leq \frac{t^2}{2} \left(\frac{r}{n}\right)^2.$$

Note, that $P(r) \leq 1/8$ if $n \geq 2tr$. Let $P_+(r)$ be the probability that the number of red balls increased in this iteration. For this to happen, at least two red balls had to be in the sample, and the deleted ball must be blue. Let $P_-(r)$ be the probability that the number of red balls decreases – the player needs to pick strictly less than two red balls in the sample, and delete a red ball. This implies

$$P_+(r) = P(r)(1 - r/n) \leq P(r) \qquad \text{and} \qquad P_-(r) = (1 - P(r))(r/n).$$

The probability for a change in the number of red balls at this iteration is

$$P_\pm(r) = P_+(r) + P_-(r) = P(r)(1 - r/n) + (1 - P(r))(r/n) = (1 - 2r/n)P(r) + r/n.$$

▶ **Lemma 10.** *Let $\xi \in (0, 1/4)$ be a parameter. If $r \leq R$, then*

$$P_+(r) \leq \frac{1/2 - \xi}{1/2 + \xi} P_-(r), \qquad \textit{where} \qquad R = (1 - 2\xi)\frac{n}{t^2}.$$

To simplify exposition, we choose $\xi = 1/6$ so that $P_+(r) \leq P_-(r)/2$. In the remaining analysis, the proofs involving tedious calculations are given in the full version of this paper [4], with the modification that the statement of the lemmas and proofs involve the parameter ξ.

What are we analyzing?

The value R/n is an upper bound on the ratio of red balls that the urn can have and still, with good probability, end with zero red balls at the end of the game. If this ratio is violated anytime during the game, then the urn might end up consisting of only red balls. We want to start the game with an urn initially having close to R red balls, but still end up with an entirely blue urn with sufficiently high probability.

The question

Let $\vartheta \in (0,1)$ and $r_0 = (1-\vartheta)R$ be the number of red balls in the urn at the start of the game (note that both r_0 and R are functions of n and t). Let $\varphi > 0$ be a parameter. The key question is the following: How large does n need to be so that if we start with r_0 red balls (and $n - r_0$ blue balls), the game ends with all balls being blue with probability $\geq 1 - \varphi$?

The game as a random walk

An iteration of the game where the number of red balls changes is an **effective** iteration. Considering only the effective iterations, this can be interpreted as a random walk starting at $X_0 = (1-\vartheta)R$ and at every iteration either decreasing the value by one with probability at least $2/3$, and increasing the value with probability at most $1/3$ (since $P_+(r) \leq P_-(r)/2$). This walk ends when either it reaches 0 or R. If the walks reaches R, then the process fails. Otherwise if the walk reaches 0, then there are no red balls in the urn, as desired.

3.2.1 Analyzing the related walk

Consider the random walk that starts at $Y_0 = (1-\vartheta)R$. In the ith iteration, $Y_i = Y_{i-1} - 1$ with probability $2/3$ and $Y_i = Y_{i-1} + 1$ with probability $1/3$. Let $\mathcal{Y} = Y_1, Y_2, \ldots$ be the resulting random walk which stochastically dominates the walk $X_0, X_1, \ldots$. This walk is strongly biased towards going to 0, and as such it does not hang around too long before moving on, as testified by the following lemma.

▶ **Lemma 11.** *Let I be any integer number, and let $\varphi > 0$ be a parameter. The number of times the random walk $\mathcal{Y}$ visits I is at most $9\ln(9/\varphi)$ times, and this holds with probability $\geq 1 - \varphi$.*

We next bound the probability that the walk fails.

▶ **Lemma 12.** *Let $\varphi > 0$ be a parameter. If $R \geq \dfrac{3}{\vartheta}\ln\dfrac{9}{\varphi}$ then the probability that the random walk ever visits R (and thus fails) is bounded by φ.*

3.2.2 Back to the urn

The number of red balls in the urn is stochastically dominated by the random walk above. The challenge is that the number of iterations one has to play before an effective iteration happens (thus, corresponding to one step of the above walk), depends on the number of red balls, and behaves like the coupon collector problem. Specifically, if there are $r \leq R$ red balls in the urn, then the probability for an effective step is $P_\pm(r) \geq (1 - P(r))(r/n) \geq r/2n$, as $P(r) \leq 1/2$. This implies that, in expectation, one has to wait at most $2n/r$ iterations before an effective iteration happens.

▶ **Lemma 13.** *Let $\varphi > 0$ be the probability of failure. For any value $r \leq R$, the urn spends at most $O\big((n/r)\log(\varphi^{-1})\big)$ regular iterations, throughout the game, having exactly r balls in it, with probability $\geq 1 - \varphi$.*

▶ **Lemma 14.** *Let $\varphi > 0$ and $\vartheta \in (0,1)$ be parameters, and assume that $n = \Omega\left(\dfrac{t^2}{\vartheta}\ln\dfrac{1}{\varphi}\right)$. The total number of regular iterations one has to play till the urn contains only blue balls, is $O\big(n\log n\log(n\varphi^{-1})\big)$, and this bound holds with probability $\geq 1 - \varphi$.*

3.3 Approximating a centerpoint

3.3.1 The algorithm

Before describing the algorithm, we need the following well known facts [9]:

(a) **Radon's theorem**: Given a set T of $d+2$ points in $\mathbb{R}^d$, one can partition T into two non-empty sets T_1, T_2, such that $\mathcal{CH}(T_1) \cap \mathcal{CH}(T_2) \neq \emptyset$. A point $p \in \mathcal{CH}(T_1) \cap \mathcal{CH}(T_2)$ is a **Radon point**.

(b) Computing a Radon point can be done by solving a system of $d+1$ linear equalities in $d+2$ variables. This can be completed in $O(d^3)$ time using Gaussian elimination.

(c) A Radon point is a $2/(d+2)$-centerpoint of T.

(d) Let h^+ be a halfspace containing only one point of T. Then, a Radon point p of T is contained in $\mathbb{R}^d \setminus h^+$ [1].

The algorithm in detail

Let P be a set of n points in $\mathbb{R}^d$ for which we would like to approximate its centerpoint. To this end, let Q be initially P. In each iteration the algorithm randomly picks $d+2$ points (with repetition) from Q, computes their Radon point, randomly deletes any point of Q, and inserts the new Radon point into the point set Q. The claim is that after a sufficient number of iterations, any point of Q is a $f(d)$-centerpoint of P, where $f(d) = \Theta(1/d^2)$ (its exact value is specified below in Eq. (3.1)).

▶ Remark 15. The algorithm above is a variant of the algorithm of Clarkson et al. [1]. Their algorithm worked in rounds, in each round generating n new Radon points, and then replacing the point set with this new set, repeating this sufficient number of times. Our algorithm on the other hand is a "continuous" process.

Intuition

A Radon point is a decent center for the points defining it. Visually, the above algorithm causes the points to slowly migrate towards the center region of the original point set.

To see why this is true, pick an arbitrary halfspace h^+ that contains exactly $f(d)n$ points of P. In each iteration, only if we picked two (or more) points that are in $Q \cap h^+$, the new point might be in h^+. Observe that we are in the setting of Radon's urn with $t = d+2$. Indeed, color all the points inside h^+ as red, and all the points outside as blue. To apply the Radon's urn analysis above, we require that $(1-\vartheta)R = f(d)n$, which by the choice of $R = (1-2\xi)n/t^2$ in Lemma 10 (and recalling $\xi = 1/6$) implies that

$$(1-\vartheta)\frac{2n}{3(d+2)^2} = f(d)n \iff f(d) = \frac{2(1-\vartheta)}{3(d+2)^2} \geq \frac{1-\vartheta}{2(d+2)^2} \tag{3.1}$$

where $\vartheta \in (0,1)$. We can now apply the Radon's urn analysis to argue that after sufficient number of iterations, all the points of Q are outside h^+. Naturally, we need to apply this analysis to all halfspaces.

3.3.2 Analysis

Consider all half-spaces that might be of interest. To this end, consider any hyperplane passing through d points of P, and translate it so that it contains on one of its sides exactly $f(d)n$ points (naturally, the are two such translations). Each such hyperplane thus defines two natural halfspaces. Let H be the resulting set of halfspaces. Observe that

$|H| \le 2\binom{n}{d} \le 2(ne/d)^d$. If Q does not contain any point in any of the halfspaces of H then all its points are $f(d)$-centerpoints. In particular, one can think about this as playing $|H|$ parallel Radon's urn games. We want the algorithm to succeed with probability $\ge 1-\varphi$. Setting the probability of success for each halfplane of H to be $\varphi/|H|$, and by Lemma 14, we have that all of these halfspaces are empty after playing

$$O\big(n \log n \log(n|H|\varphi^{-1})\big) = O\big(dn \log n \log(n/\varphi)\big)$$

iterations, with probability of success being $1 - |H|(\varphi/|H|) = 1-\varphi$ by the union bound. Using Lemma 14 requires that $n = \Omega\big(t^2\vartheta^{-1}\ln(|H|/\varphi)\big) = \Omega\big(\vartheta^{-1}d^3 \ln n + \vartheta^{-1}d^2 \ln \varphi^{-1}\big)$ which holds for $n = \Omega(\vartheta^{-1}d^3 \ln d + \vartheta^{-1}d^2 \ln \varphi^{-1})$.

Using the fact that computing a Radon point for $d+2$ points in $\mathbb{R}^d$ can be done in $O(d^3)$ time, we obtain the following result.

▶ **Lemma 16.** *Let $\varphi > 0$ and $\vartheta \in (0,1)$ be parameters, and let P be a set of $n = \Omega(\vartheta^{-1}d^3 \ln d + \vartheta^{-1}d^2 \ln \varphi^{-1})$ points in $\mathbb{R}^d$. Let $\alpha = \frac{1-\vartheta}{2(d+2)^2}$. Then, one can compute a α-centerpoint of P via a randomized algorithm. The running time of the randomized algorithm is $O\big(d^3 \cdot dn \log n \log(n/\varphi)\big) = O(d^4 n \log n \log(n/\varphi))$, and it succeeds with probability $\ge 1-\varphi$.*

▶ **Theorem 17.** *Given a set P of n points in $\mathbb{R}^d$, a parameter φ, and a constant $\vartheta \in (0,1)$, one can compute a $\frac{1-\vartheta}{2(d+2)^2}$-centerpoint of P. The running time of the algorithm is $O\big(\vartheta^{-3}d^7 \log^3 d \log^3 \varphi^{-1}\big)$, and it succeeds with probability $\ge 1-\varphi$.*

Proof. The idea is to pick a random sample S from P that is a $(\rho, \vartheta/8)$-relative approximation for halfspaces, where $\rho = 1/(10d^2)$. This range space has VC dimension $d+1$, and by Theorem 7, a sample of size

$$\mu = O\big(\rho^{-1}\vartheta^{-2}(d \log \rho^{-1} + \log \varphi^{-1})\big) = O\big(d^2\vartheta^{-2}(d \log d + \log \varphi^{-1})\big)$$

is a $(\rho, \vartheta/8)$-relative approximation.

Running the algorithm of Lemma 16 on S with $\vartheta/8$ yields a τ-centerpoint $\bar{c}$ of S, where $\tau = \frac{1-\vartheta/8}{2(d+2)^2} \ge \rho$ for $d \ge 2$. By the relative approximation property, this is a $(1 \pm \vartheta/8)\tau$-centerpoint of P. Therefore $\bar{c}$ is an α-centerpoint for P, where

$$\alpha = (1-\vartheta/8)\tau = \frac{(1-\vartheta/8)^2}{(d+2)^2} \ge \frac{1-\vartheta}{(d+2)^2}.$$

By Lemma 16, the running time of the resulting algorithm is

$$O(d^4 \mu \log \mu \log(\mu/\varphi)) = O\big(\vartheta^{-2}\log^2\vartheta^{-1}d^7 \log^3 d \log^3 \varphi^{-1}\big) = O\big(\vartheta^{-3}d^7 \log^3 d \log^3 \varphi^{-1}\big). \blacktriangleleft$$

Now, one can repeat the above calculations with the parameter ξ. Intuitively, as ξ approaches zero, the random walk becomes less unbalanced since it will move left with probability $1/2+\xi$ and right with probability $1/2-\xi$. Because of this, there is an increased chance that the random walk will reach R (and thus fail). In order to preserve that the random walk succeeds with probability at least $1-\varphi$, the sample size n must depend on the parameter ξ. In fact, the parameter ξ allows us to compute centerpoints with quality arbitrarily close to $1/(d+2)^2$.

▶ **Theorem 18.** *Given a set P of n points in $\mathbb{R}^d$, a parameter φ, and a constant $\gamma \in (0,1)$, one can compute a $\frac{1-\gamma}{(d+2)^2}$-centerpoint of P. The running time of the algorithm is $O\big(\gamma^{-4}d^7 \log^3 d \log^3(\gamma^{-1}\varphi^{-1})\big)$, and it succeeds with probability $\ge 1-\varphi$.*

▶ Remark 19. The above compares favorably to the result of [1, Corollary 3] – they get a running time of $O(d^9 \log d + d^8 \log^2 \varphi^{-1})$, which is slower by roughly a factor of d^2, and computes a $\frac{1}{4.08(d+2)(d+1)}$-centerpoint of P – the quality of the centerpoint is roughly worse by a factor of four.

4 Application I: Algorithms with oracle access

In this section we discuss two applications of the improved centerpoint algorithm. Both applications revolve around the idea of *oracle access*. In the first application, we are interested in lower bounding a convex function given an oracle to compute it's gradient. In the second, we utilize centerpoints in order to determine whether a given convex body C is ε-heavy using a separation oracle.

4.1 Lowerbounding a function with a gradient oracle

▶ **Definition 20.** *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex function. For a number $c \in \mathbb{R}$, define the* ***level set of*** *f as $\mathcal{L}_f(c) = \left\{ p \in \mathbb{R}^d \;\middle|\; f(p) \leq c \right\}$. If f is a convex function, then $\mathcal{L}_f(c)$ is a convex set for all $c \in \mathbb{R}$.*

▶ **Definition 21.** *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex (and possibly non-differentiable) function. For a point $p \in \mathbb{R}^d$, a vector $v \in \mathbb{R}^d$ is a* ***subgradient*** *of f at p if for all $q \in \mathbb{R}^d$, $f(q) \geq f(p) + \langle v, q - p \rangle$. The* ***subdifferential*** *of f at $p \in \mathbb{R}^d$, denoted by $\partial f(p)$, is the set of all subgradients $v \in \mathbb{R}^d$ of f at p. When the domain of f is $\mathbb{R}^d$ and f is convex, then $\partial f(p)$ is a non-empty set for all $p \in \mathbb{R}^d$.*[1]

▶ **Theorem 22.** *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex (possibly non-differentiable) function and P a set of n points in $\mathbb{R}^d$. Assume that one has access to an oracle which given $p \in \mathbb{R}^d$ returns an arbitrary element in the subdifferential $\partial f(p)$. With $O(d^2 \log n)$ queries to the oracle, one can compute a point $q \in \mathbb{R}^d$ (not necessarily in P) such that $f(q) \leq \min_{p \in P} f(p)$.*

Proof. Let $P_1 = P$, and $P_i \subseteq P$ denote the set of remaining points at the beginning of the ith iteration. In iteration i, for some constant $c > 0$, compute a (c/d^2)-centerpoint $\overline{c}_i$ of P_i using Theorem 17 in time $O(d^7 \log^3 d)$ with success probability $1/2$. Define $\mathsf{C}_i = \mathcal{L}_f(f(\overline{c}_i))$. We now use the oracle to obtain subgradient vector $v \in \partial f(\overline{c}_i)$. Using v, we obtain a d-dimensional hyperplane h_i which is tangent to C_i at $\overline{c}_i$. Let h_i^+ be the halfspace formed from h_i which contains the interior of C_i. If $\left| h_i^- \cap P_i \right| \geq c \left| P_i \right| / d^2$, then such an iteration is successful and we set $P_{i+1} = P_i \setminus (h_i^- \cap P_i)$ and continue to iteration $i + 1$. Otherwise the iteration has failed and we repeat the ith iteration. This procedure is repeated until we reach an iteration τ in which $|P_\tau|$ is of constant size. At this stage, the algorithm returns the point achieving the minimum of $\min_{1 \leq i \leq \tau} f(\overline{c}_i)$ and $\min_{p \in P_\tau} f(p)$. Because f is convex, the algorithm returns a point q such that $f(q) \leq f(p)$ for all $p \in P$.

As for the number of queries, note that in each iteration the expected number of centerpoint calculations (and thus queries) until a successful iteration is $O(1)$. It remains to bound the number of successful iterations. In each successful iteration, a c/d^2-fraction of points are discarded. Therefore there are at most τ iterations, for which τ is the smallest number with $(1 - c/d^2)^\tau n$ smaller than some constant. This implies $\tau = O(d^2 \log n)$. ◀

[1] Indeed, consider the epigraph of the function f, $S = \left\{ (x, c) \;\middle|\; x \in \mathbb{R}^d, c \in \mathbb{R}, f(x) \leq c \right\} \subseteq \mathbb{R}^{d+1}$. Observe that S is a convex set. For a given point $p \in \mathbb{R}^d$, by the supporting hyperplane theorem, there is some hyperplane h tangent to S at the point $(p, f(p)) \in \mathbb{R}^{d+1}$. The normal to this tangent hyperplane h is one possible subgradient of f at p.

4.2 Functional nets: A weak net in the oracle model

4.2.1 The model, construction, and query process

Model

Given a convex body $\mathsf{C} \subseteq \mathbb{R}^d$, we assume oracle access to it. This is a standard model in optimization. Specifically, given a query point q, the oracle either returns that $q \in \mathsf{C}$, or alternatively it returns a (separating) hyperplane h, such that C lies completely on one side of h, and q lies on the other side.

Our purpose here is to precompute a small subset $S \subseteq P$, such that given any convex body C (with oracle access to it), one can decide if C is ε-light. Specifically, the query algorithm (using only S, and not the whole point set P) generates an (adaptive) sequence of query points $q_1, q_2, \ldots$, such that if any of these query points are in C, then the algorithm considers C to be heavy. Otherwise, if all the query points miss C, then the algorithm outputs (correctly) that C is light (i.e., $\overline{m}(\mathsf{C}) < \varepsilon$).

Construction

Given P, the set S is a random sample from P of size

$$\mu = O\bigl(\varepsilon^{-1} d^3 \log d \log^3 \varepsilon^{-1} + \varepsilon^{-1} \log \varphi^{-1}\bigr) = \widetilde{O}\bigl(d^3/\varepsilon\bigr), \tag{4.1}$$

where $\varphi > 0$ is a prespecified parameter.

Query process

Given a convex body C (with oracle access to it), the algorithm starts with $S_0 = S$. In the ith iteration, the algorithm computes a $\Omega(1/d^2)$-centerpoint q_i of S_i using the algorithm of Theorem 17, with failure probability at most $1/4$. If the oracle returns that $q_i \in \mathsf{C}$, then the algorithm returns q_i as a proof of why C is considered to be heavy. Otherwise, the oracle returns a separating hyperplane h_i, such that the open halfspace h_i^- contains q_i. Let $S_i' = S_{i-1} \setminus h_i^-$. If $|S_i'| \leq (1-\gamma)\,|S_{i-1}|$, where $\gamma = 1/16d^2$ then we set $S_i = S_i'$ (such an iteration is called **successful**). Otherwise, we set $S_i = S_{i-1}$. The algorithm stops when $|S_i| \leq \varepsilon\,|S|/8$.

4.2.2 Correctness

Let I be the set of indices of all the successful iterations, and consider the convex set $\mathsf{C}_I = \cap_{i \in I} h_i^+$. The set C_I is an outer approximation to C. In particular, for an index j, let $\mathsf{C}_j = \cap_{i \in I, i \leq j} h_i^+$ be this outer approximation in the end of the jth iteration. We have that $S_j = S \cap \mathsf{C}_j$.

The proofs of the following results can be found in the full version of the paper [4].

▶ **Lemma 23.** *There are at most $\tau = O(d^2 \log \varepsilon^{-1})$ successful iterations. For any j, the convex polyhedron C_j is defined by the intersection of at most τ closed halfspaces.*

Let $\mathcal{H}^\tau$ be the set of all of convex polyhedra in $\mathbb{R}^d$ that are formed by the intersection of τ closed halfspaces.

▶ **Observation 24.** *The VC dimension of $(\mathbb{R}^d, \mathcal{H}^\tau)$ is*

$$D = O(d\tau \log \tau) = O\bigl(d\bigl(d^2 \log \varepsilon^{-1}\bigr) \log\bigl(d^2 \log \varepsilon^{-1}\bigr)\bigr) = O(d^3 (\log d) \log^2 \varepsilon^{-1}).$$

This follows readily, as the VC dimension of the range space of points in $\mathbb{R}^d$ and halfspaces is $d+1$, and by the bound of Lemma 8 for the intersection of τ such ranges.

▶ **Lemma 25.** *The set S is a relative $(\varepsilon/8, 1/4)$-approximation for $(P, \mathcal{H}^\tau)$, with probability $1 - \varphi$.*

Proof. Using Theorem 7 with $p = \varepsilon/8$, $\widehat{\varepsilon} = 1/4$, and $\xi = D$, implies that a random sample of P of size

$$O\bigl(\widehat{\varepsilon}^{-2} p^{-1} \bigl(\xi \log p^{-1} + \log \varphi^{-1}\bigr)\bigr) = O\bigl(\varepsilon^{-1}\bigl(D \log \varepsilon^{-1} + \log \varphi^{-1}\bigr)\bigr)$$
$$= O\bigl(\varepsilon^{-1}\bigl(d^3 \log d \log^3 \varepsilon^{-1} + \log \varphi^{-1}\bigr)\bigr)$$

is the desired relative $(p, \widehat{\varepsilon})$-approximation with probability $\geq 1 - \varphi$. And this is indeed the size of S, see Eq. (4.1). ◀

▶ **Lemma 26.** *Given a convex query body C, the expected number of oracle queries performed by the algorithm is $O(d^2 \log \varepsilon^{-1})$, and the expected running time of the algorithm is $O(d^9 \varepsilon^{-1} \mathrm{polylog})$, where $\mathrm{polylog} = O\bigl(\log(d\varepsilon^{-1}\varphi^{-1})^{O(1)}\bigr)$.*

▶ **Lemma 27.** *Assuming that S is the desired relative approximation, then for any query body C, if the algorithm declares that it is ε-light, then $|\mathsf{C} \cap P| < \varepsilon n$.*

The above implies the following.

▶ **Theorem 28.** *Let P be a set of points in $\mathbb{R}^d$, and let $\varepsilon, \varphi > 0$ be parameters. Let S be a random sample of P of size*

$$\mu = O\bigl(\varepsilon^{-1} d^3 \log d \log^3 \varepsilon^{-1} + \varepsilon^{-1} \log \varphi^{-1}\bigr) = \widetilde{O}(\varepsilon^{-1} d^3).$$

Then, for a given query convex body C endowed with an oracle access, the algorithm described above, which uses only S, computes a sequence of query points $q_1, \ldots, q_m$, such that either:

(i) *one of the points $q_i \in \mathsf{C}$, and the algorithm outputs q_i as a "proof" that C is ε-heavy, or*
(ii) *the algorithm outputs that $|\mathsf{C} \cap P| < \varepsilon n$.*

The query algorithm has the following performance guarantees:

(a) *The expected number of oracle queries is $\mathbf{E}[m] = O(d^2 \log \varepsilon^{-1})$.*
(b) *The algorithm itself (ignoring the oracle queries) runs in $\widetilde{O}\bigl(d^9 \varepsilon^{-1}\bigr)$ time*

The output of the algorithm is correct, for all convex bodies, with probability $\geq 1 - \varphi$.

▶ Remark 29. One may hope to bound the probability of the algorithm reporting a false positive. However this is inherently not possible for any weak ε-net construction. Indeed, the algorithm can fail to distinguish between a polygon that contains at least εn of the points of P and a polygon that contains *none* of the points of P. Consider n points P lying on a circle in $\mathbb{R}^2$. Choose εn of these points on the circle, and let C be the convex hull of these points. Clearly C contains at least εn points of P. Now, take each vertex in C and "slice" it off, forming a new polygon C' that contains no points from P. However, C' is still a large polygon and as such may contain a centerpoint during the execution of the above algorithm. Therefore our algorithm may report that C' contains a large fraction of the points, even though C' is contains no points of P, and so it fails to distinguish between C and C'.

▶ Remark 30. Clarkson et al. [1] provide also a randomized algorithm that finds a $\bigl(\frac{1}{d+1} - \gamma\bigr)$-centerpoint with probability $1 - \delta$ in time $O\bigl(\bigl[d\gamma^{-2} \log(d\gamma^{-1})\bigr]^{d+O(1)} \log \delta^{-1}\bigr)$. We could use this algorithm instead of Theorem 17 in the query process. Since we are computing a better quality centerpoint, the number of iterations τ and sample size μ would be smaller by a factor of d. Specifically, $\tau = O(d \log \varepsilon^{-1})$ and from Lemma 8, the VC dimension of the range space $\mathsf{S} = (P, \mathcal{H}^\tau)$ becomes $D = O(d^2 \log d \log^2 \varepsilon^{-1})$. Following the proof of Lemma 25, we can construct a sample S which is $(\varepsilon/8, 1/4)$-relative approximation for S with probability $1-\varphi$ of size

$$\mu = O\big(\varepsilon^{-1}(D \log \varepsilon^{-1} + \log \varphi^{-1})\big) = O\big(\varepsilon^{-1}(d^2 \log d \log^3 \varepsilon^{-1} + \log \varphi^{-1})\big). \tag{4.2}$$

5 Application II: Constructing center nets

We next introduce a strengthening of the concept of a weak ε-net. Namely, we require that there is a point p in the net which stabs an ε-heavy convex body C, and that p is also a good centerpoint for $\mathsf{C} \cap P$.

▶ **Definition 31.** *For a set P of n points in $\mathbb{R}^d$, and parameters $\varepsilon, \alpha \in (0,1)$, a subset $\mathcal{W} \subseteq \mathbb{R}^d$ is an (ε, α)-**center net** if for any convex shape C, such that $|P \cap \mathsf{C}| \geq \varepsilon n$, we have that there is an α-centerpoint of $P \cap \mathsf{C}$ in $\mathcal{W}$.*

In this section we prove existence of an (ε, α)-center net $\mathcal{W}$ of size roughly $O_d(\varepsilon^{-d^2})$, where

$$\alpha = \frac{\mathsf{c}_1}{(d+1)\log \varepsilon^{-1}},$$

and $\mathsf{c}_1 \in (0,1)$ is some fixed constant to be specified shortly. Note that the quality of the centerpoint is worse by a factor of $\log \varepsilon^{-1}$ than the best one can hope for.

5.1 The construction

The construction of the center net will be based an algorithm for constructing a weak ε-net for P. In particular, the construction algorithm will use the following two results (the proofs can be found in the full version of the paper [4]).

▶ **Lemma 32.** *Given a set P of n points in $\mathbb{R}^d$, one can compute a set Q of $O\big(n^{d^2}\big)$ points, such that for any subset $P' \subseteq P$, there is a $1/(d+1)$-centerpoint of P' in Q.*

▶ **Lemma 33.** *Let P be a set of n points in $\mathbb{R}^d$. Let S be a random sample from P of size $\mu = \widetilde{O}(\varepsilon^{-1} d^2)$, see Eq. (4.2) for the exact bound. Then, one can compute a set of points $\mathcal{W}$ from S, of size*

$$O(\mu^{d^2}) = O\Big(\big(\varepsilon^{-1}(d^2 \log d \log^3 \varepsilon^{-1} + \log \varphi^{-1})\big)^{d^2}\Big)$$

which is a weak ε-net for P with probability $\geq 1-\varphi$.

▶ Remark 34. A similar construction of a weak ε-net, to the one in Lemma 33, from a small sample was described by Mustafa and Ray [12]. Their sample has exponential dependency on the dimension, so the resulting weak ε-net has somewhat worse dependency on the dimension than our construction. In any case, these constructions are inferior as far as the dependency on ε, compared to the work of Matoušek and Wagner [10], which shows a weak ε-net construction of size $O_d(\varepsilon^{-d}(\log \varepsilon^{-1})^{O(d^2 \log d)})$.

The idea will be to repeat the construction of the net of Lemma 33, with somewhat worse constants. Specifically, take a sample S of size $\mu = \widetilde{O}(\varepsilon^{-1} d^2)$ from P, see Eq. (4.2) for the exact bound. Next, we construct the set $\mathcal{W}$ for S, using the result of Lemma 32. Return $\mathcal{W}$ as the desired (ε, α)-center net.

5.2 Correctness

The proof is algorithmic. Fix any convex ε-heavy body C, and let $S_1 = S$ be the **active set** and let $P_1 = \mathsf{C} \cap P$ be the **residual set** in the beginning of the first iteration.

We now continue in a similar fashion to the algorithm of Theorem 28. In the ith iteration, the algorithm computes the $1/(d+1)$-centerpoint q_i of S_i (running times do not matter here, so one can afford computing the best possible centerpoint). If q_i is a 2α-centerpoint for P_i, then q_i is intuitively a good centerpoint for P, and the algorithm returns q_i as the desired center point. Observe that by construction, $q_i \in \mathcal{W}$ as desired.

If not, then there exists a closed halfspace h_i^+ containing q_i and at most $2\alpha\,|P_i|$ points of P_i. Let

$$P_{i+1} = P_i \setminus h_i^+ \qquad \text{and} \qquad S_{i+1} = S_i \setminus h_i^+.$$

The algorithm now continues to the next iteration.

Analysis

The key insight is that the active set S_i shrinks much faster than the residual set P_i. However, by construction, S_i provides a good upper bound to the size of P. Now once the upper bound provided by S_i on the size of P_i is too small, this would imply that the algorithm must have stopped earlier, and found a good centerpoint.

▶ **Lemma 35.** *Let $\tau = \left\lceil 1 + 3(d+1) + (d+1)\log \varepsilon^{-1} \right\rceil$, and $\alpha = 1/(4\tau)$. Assuming that S is a relative $(\varepsilon/8, 1/4)$-approximation for the range space $\mathsf{S} = (P, \mathcal{H}^\tau)$, the above algorithm stops after at most τ iterations.*

▶ **Lemma 36.** *The above algorithm outputs a α-centerpoint of $P \cap \mathsf{C}$.*

Arguing as in Remark 30 implies the following.

▶ **Corollary 37.** *For the above algorithm to succeed with probability $\geq 1 - \varphi$, the sample S needs to be a sample of the size specified by Eq. (4.2).*

5.3 The result

▶ **Theorem 38.** *Let P be a set of n points in $\mathbb{R}^d$, and $\varepsilon > 0$ be a parameter. For $\gamma = \log(1/\varepsilon)$, there exists a $\big(\varepsilon, \Omega(1/(d\gamma))\big)$-center net $\mathcal{W}$ (which is also a weak ε-net) of P (see Definition 31). The size of the net $\mathcal{W}$ is $O(\mu^{d^2}) \approx O_d(\varepsilon^{-d^2})$, where $\mu = \widetilde{O}(\varepsilon^{-1} d^2)$, see Eq. (4.2) for the exact bound.*

Proof. The theorem follows readily from the above, by setting $\varphi = 1/2$. ◀

References

1. K. L. Clarkson, D. Eppstein, G. L. Miller, C. Sturtivant, and S.-H. Teng. Approximating center points with iterative Radon points. *Internat. J. Comput. Geom. Appl.*, 6:357–377, 1996. URL: `http://cm.bell-labs.com/who/clarkson/center.html`.
2. S. Har-Peled. *Geometric Approximation Algorithms*, volume 173 of *Math. Surveys & Monographs*. Amer. Math. Soc., Boston, MA, USA, 2011. `doi:10.1090/surv/173`.
3. S. Har-Peled and M. Sharir. Relative (p, ε)-Approximations in Geometry. *Discrete Comput. Geom.*, 45(3):462–496, 2011. `doi:10.1007/s00454-010-9248-1`.
4. Sariel Har-Peled and Mitchell Jones. Journey to the Center of the Point Set. *CoRR*, abs/1712.02949, 2019. `arXiv:1712.02949`.

5 D. Haussler and E. Welzl. ε-nets and simplex range queries. *Discrete Comput. Geom.*, 2:127–151, 1987. doi:10.1007/BF02187876.

6 C. Keller, S. Smorodinsky, and G. Tardos. Improved bounds on the Hadwiger-Debrunner numbers. *ArXiv e-prints*, December 2015. arXiv:1512.04026.

7 C. Keller, S. Smorodinsky, and G. Tardos. On Max-Clique for intersection graphs of sets and the Hadwiger-Debrunner numbers. In Philip N. Klein, editor, *Proc. 28th ACM-SIAM Sympos. Discrete Algs.* (SODA), pages 2254–2263. SIAM, 2017. doi:10.1137/1.9781611974782.148.

8 Y. Li, P. M. Long, and A. Srinivasan. Improved Bounds on the Sample Complexity of Learning. *J. Comput. Syst. Sci.*, 62(3):516–527, 2001.

9 J. Matoušek. *Lectures on Discrete Geometry*, volume 212 of *Grad. Text in Math.* Springer, 2002. doi:10.1007/978-1-4613-0039-7/.

10 J. Matoušek and U. Wagner. New Constructions of Weak epsilon-Nets. *Discrete Comput. Geom.*, 32(2):195–206, 2004. URL: http://www.springerlink.com/index/10.1007/s00454-004-1116-4.

11 G. L. Miller and D. R. Sheehy. Approximate centerpoints with proofs. *Comput. Geom.*, 43(8):647–654, 2010. doi:10.1016/j.comgeo.2010.04.006.

12 N. H. Mustafa and S. Ray. Weak ε-nets have basis of size $O(\varepsilon^{-1} \log \varepsilon^{-1})$ in any dimension. *Comput. Geom. Theory Appl.*, 40(1):84–91, 2008. doi:10.1016/j.comgeo.2007.02.006.

13 N. H. Mustafa and K. Varadarajan. Epsilon-approximations and epsilon-nets. *CoRR*, abs/1702.03676, 2017. arXiv:1702.03676.

14 A. Rok and S. Smorodinsky. Weak $1/r$-Nets for Moving Points. In *Proc. 32nd Int. Annu. Sympos. Comput. Geom.* (SoCG), pages 59:1–59:13, 2016. doi:10.4230/LIPIcs.SoCG.2016.59.

15 N. Rubin. An Improved Bound for Weak Epsilon-Nets in the Plane. In *Proc. 59th Annu. Sympos. on Found. of Comput. Sci.,* (FOCS), pages 224–235, 2018. doi:10.1109/FOCS.2018.00030.

16 V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.*, 16:264–280, 1971.