Tighter Estimates for -nets for Disks

Norbert Bus, Shashwat Garg, Nabil Mustafa, Saurabh Ray

Tighter Estimates for ϵ-nets for Disks

Norbert Bus
Université Paris-Est,
Laboratoire d'Informatique Gaspard-Monge,
Equipe A3SI, ESIEE Paris.
busn@esiee.fr

Shashwat Garg
IIT Delhi
garg.shashwat@gmail.com

Nabil H. Mustafa*
Université Paris-Est,
Laboratoire d'Informatique Gaspard-Monge,
Equipe A3SI, ESIEE Paris.
mustafan@esiee.fr

Saurabh Ray
Computer Science, New York University, Abu Dhabi.
saurabh.ray@nyu.edu

Abstract

The geometric hitting set problem is one of the basic geometric combinatorial optimization problems: given a set P of points, and a set $\mathcal{D}$ of geometric objects in the plane, the goal is to compute a small-sized subset of P that hits all objects in $\mathcal{D}$. In 1994, Bronniman and Goodrich [5] made an important connection of this problem to the size of fundamental combinatorial structures called ϵ-nets, showing that small-sized ϵ-nets imply approximation algorithms with correspondingly small approximation ratios. Very recently, Agarwal and Pan [2] showed that their scheme can be implemented in near-linear time for disks in the plane. Altogether this gives $O(1)$-factor approximation algorithms in $\tilde{O}(n)$ time for hitting sets for disks in the plane.

This constant factor depends on the sizes of ϵ-nets for disks; unfortunately, the current state-of-the-art bounds are large – at least $24/\epsilon$ and most likely larger than $40/\epsilon$. Thus the approximation factor of the Agarwal and Pan algorithm ends up being more than 40. The best lower-bound is $2/\epsilon$, which follows from the Pach-Woeginger construction [32] for halfplanes in two dimensions. Thus there is a large gap between the best-known upper and lower bounds. Besides being of independent interest, finding precise bounds is important since this immediately implies an improved linear-time algorithm for the hitting-set problem.

The main goal of this paper is to improve the upper-bound to $13.4/\epsilon$ for disks in the plane. The proof is constructive, giving a simple algorithm that uses only Delaunay triangulations. We have implemented the algorithm, which is available as a public open-source module. Experimental results show that the sizes of ϵ-nets for a variety of data-sets is lower, around $9/\epsilon$.

*The work of Nabil H. Mustafa in this paper has been supported by the grant ANR SAGA (JCJC-14-CE25-0016-01).

1 Introduction

The minimum hitting set problem is one of the most fundamental combinatorial optimization problems: given a range space $(P, \mathcal{D})$ consisting of a set P and a set $\mathcal{D}$ of subsets of P called the *ranges*, the task is to compute the smallest subset $Q \subseteq P$ that has a non-empty intersection with each of the ranges in $\mathcal{D}$. This problem is strongly NP-hard. If there are no restrictions on the set system $\mathcal{D}$, then it is known that it is NP-hard to approximate the minimum hitting set within a logarithmic factor of the optimal [34]. The problem is NP-complete even for the case where each range has exactly two points since this problem is equivalent to the vertex cover problem which is known to be NP-complete [22, 18]. A natural occurrence of the hitting set problem occurs when the range space $\mathcal{D}$ is derived from geometry – e.g., given a set P of n points in $\mathbb{R}^2$, and a set $\mathcal{D}$ of m triangles containing points of P, compute the minimum-sized subset of P that hits all the triangles in $\mathcal{D}$. Unfortunately, for most natural geometric range spaces, computing the minimum-sized hitting set remains NP-hard. For example, even the (relatively) simple case where $\mathcal{D}$ is a set of unit disks in the plane is strongly NP-hard [21]. Therefore fast algorithms for computing provably good approximate hitting sets for geometric range spaces have been intensively studied for the past three decades (e.g., see the two recent PhD theses on this topic [16, 17]).

Computing hitting sets for disks in the plane has been the subject of a long line of research. The case when all the disks have the same radius is easier, and has been studied in a series of works: Călinsecu *et al.* [8] proposed a 108-approximation algorithm, which was subsequently improved by Ambhul *et al.* [3] to 72. Carmi *et al.* [9] further improved that to a 38-approximation algorithm, though with the running time of $O(n^6)$. Claude *et al.* [13] were able to achieve a 22-approximation algorithm running in time $O(n^6)$. More recently Fraser *et al.* [14] presented a 18-approximation algorithm in time $O(n^2)$. Mustafa *et al.* [28] showed a QPTAS for the dual problem of covering points by weighted disks and pseudo-disks in the plane.

So far, besides ad-hoc approaches, there are two systematic lines along which all progress on the hitting-set problem for geometric ranges has relied on: rounding via ϵ-nets, and local-search. The local-search approach starts with any hitting set $S \subseteq P$, and repeatedly decreases the size of S, if possible, by replacing k points of S with $\leq k-1$ points of $P \setminus S$. Call such an algorithm a k-local search algorithm. It has been shown [30] that a k-local search algorithm for the hitting set problem for disks in the plane gives a PTAS. Unfortunately the running time of their algorithm to compute a $(1+\epsilon)$-approximation is $O(n^{O(1/\epsilon^2)})$. Very recently Bus *et al.* [6] were able to improve the analysis and algorithm of the local-search approach to design a 8-approximation running in time $O(n^{2.33})$. However, at this moment, a near-linear time algorithm based on local-search seems beyond reach. We currently do not even know how to compute the most trivial case, namely when $k = 1$, of local-search in near-linear time: given the set of disks D, and a set of points P, compute a *minimal* hitting set in P of D.

Rounding via ϵ-nets. Given a range space $(P, \mathcal{D})$ and a parameter $\epsilon > 0$, an ϵ-net is a subset $S \subseteq P$ such that $D \cap S \neq \emptyset$ for all $D \in \mathcal{D}$ with $|D \cap P| \geq \epsilon n$. The famous ϵ-net theorem of Haussler and Welzl [20] states that for range spaces with VC-dimension d, there exists an ϵ-net of size $O(d/\epsilon \log d/\epsilon)$; this bound was later improved to $O(d/\epsilon \log 1/\epsilon)$ and which was shown to be optimal in general [23]). Sometimes, weighted versions of the problem are considered in which each $p \in P$ has some positive weight associated with it so that the total weight of all elements of P is 1. The weight of each range is the sum of the weights of the elements in it. The aim is to hit all ranges with weight more than ϵ. The condition of having finite VC-dimension is satisfied by many geometric set systems: disks, half-spaces, k-sided polytopes, r-admissible set of regions etc. in $\mathbb{R}^d$. For certain range spaces, one can further improve the bound of the ϵ-net theorem [36, 10, 12, 29, 24, 35, 31]. An important case is ϵ-net for disks in the plane, for which there are several proofs showing the existence of $O(1/\epsilon)$-sized nets [33].

In 1994, Bronnimann and Goodrich [5] proved the following interesting connection between the hitting-set problem, and ϵ-nets: let $(P, \mathcal{D})$ be a range-space for which we want to compute a minimum hitting set. If one can compute an ϵ-net of size c/ϵ for the ϵ-net problem for $(P, \mathcal{D})$ in polynomial time, then one can compute a hitting set of size at most $c \cdot \text{OPT}$ for $(P, \mathcal{D})$, where OPT is the size of the optimal (smallest) hitting set, in polynomial time. A shorter, simpler proof was given by Even *et al.* [15]. Both these proofs construct an assignment of weights to points in P such that the total weight of each range $D \in \mathcal{D}$ (i.e., the sum of the weights of the points in D) is at least $(1/\text{OPT})$-th fraction of the total weight. Then a $(1/\text{OPT})$-net with these weights is a hitting set. Until very recently, the best such rounding algorithms had running times of $\Omega(n^2)$, and it had been a long-standing open problem to compute a $O(1)$-approximation to the hitting-set problem for disks in the plane in near-linear time. In a recent break-through, Agarwal and Pan [2] presented an algorithm that is able to do the required rounding efficiently for a broad set of geometric objects. In particular, they are able to get the first near-linear algorithm for computing $O(1)$-approximations for hitting sets for disks.

Bounds on ϵ-nets. The result of Agarwal and Pan [2] opens the way, for the first time, for near linear-time algorithms for the geometric hitting set problem. The catch is that the approximation factor depends on the sizes of ϵ-nets for disks; despite over seven different proofs of $O(1/\epsilon)$-sized ϵ-nets for disks, the precise bounds are not very encouraging. The paper containing the earliest proof, Matousek *et al.* [27], was over twenty-two years ago and thus summarized their result:

"Note that in principle the ϵ-net construction presented in this paper can be transformed into a deterministic algorithm that runs in polynomial time, $O(n^3)$ at worst. However, we certainly would not advocate this algorithm as being practical. We find the resulting constant of proportionality also not particularly flattering." [27]

So far, the best constants for the ϵ-nets come from the proofs in [33] and [19]. Denote by $f(\alpha)$ the best known bound on the size of an α-net for lower halfspaces in $\mathbb{R}^3$. A lifting of the problem of disks to $\mathbb{R}^3$ gives an ϵ-net problem with lower halfspaces in $\mathbb{R}^3$. The former paper constructs $\frac{1}{4}$-nets for $4/\epsilon$ independent sub-problems, resulting in ϵ-nets of size $\frac{4}{\epsilon} f(\frac{1}{4})$ for halfspaces in $\mathbb{R}^3$. The latter paper presents five proofs for the existence of linear size ϵ-nets for halfspaces in $\mathbb{R}^3$. The best constant for disks is obtained by using their first proof, obtaining a bound of $\frac{4}{\epsilon} f(\alpha)$ where $\alpha < \frac{1}{3}$. Thus, by using the lower bound of [32] for halfplanes in $\mathbb{R}^2$, $f(\alpha) \geq \lceil 2/\alpha \rceil - 1$, the best constructions give a bound that is at least $24/\epsilon$. However, we believe that $f(\alpha) \geq 10$ for $\alpha <= \frac{1}{2}$ since even for $\epsilon = 1/2$, no ϵ-net construction of size less than 10 is known. Such bounds would result in ϵ-nets of size at least $40/\epsilon$. Furthermore, there is no implementation or software solution available that can even compute such ϵ-nets efficiently.

Our Contributions

We prove new improved bounds on sizes of ϵ-nets and present efficient algorithms to compute such nets. Our approach is simple: we will show that modifications to a well-known technique for computing ϵ-nets – the sample-and-refine approach of Chazelle-Friedman [11] – together with additional structural properties of Delaunay triangulations in fact results in ϵ-nets of surprisingly low size:

Theorem 1.1. *Given a set P of n points in $\mathbb{R}^2$, there exists an ϵ-net under disk ranges of size at most $13.4/\epsilon$. Furthermore it can be computed in expected time $O(n \log n)$.*

A major advantage of Delaunay triangulations is that their behavior has been extensively studied, there are many efficient implementations available, and they exhibit good behavior for various real-world data-sets as well as random point sets. The algorithm, using CGAL, is furthermore simple to implement. We have implemented it, and present the sizes of ϵ-nets for various real-world data-sets; the results indicate that our theoretical analysis closely tracks the actual size of the nets. This can additionally be seen as continuing the program for better analysis of basic geometric tools; see, e.g., Matousek [25] for a detailed analysis for a related structure called cuttings.

Together with the result of Agarwal and Pan, this immediately implies the following:

Corollary 1.1. *For any $\delta > 0$, one can compute a $(13.4 + \delta)$-approximation to the minimum hitting set for $(P, \mathcal{D})$ in time $\tilde{O}(n)$.*

2 A near linear time algorithm for computing ϵ-nets for disks in the plane

Through a careful analysis, we present an algorithm for computing an ϵ-net of size $\frac{13.4}{\epsilon}$, running in near linear time. The method, shown in Algorithm 1, computes a random sample and then solves certain sub-problems involving subsets of points located in pairs of Delaunay disks circumscribing adjacent triangles in the Delaunay triangulation of the random sample. The main idea is that every disk that is not hit by a point in the random sample is contained in such a sub-problem. The key to our bounds is $i)$ considering edges in the Delaunay triangulation instead of faces in the analysis, and $ii)$ new improved constructions for large values of ϵ. We note that further improvements on the latter would probably yield smaller ϵ-nets.

Let $\Delta(abc)$ denote the triangle defined by the three points a, b and c. D_{abc} denotes the disk through a, b and c, while $D_{ab\bar{c}}$ denotes the halfplane defined by a and b not containing the point c. Let $c(D)$ denote the center of the disk D.

Let $\Xi(R)$ be the Delaunay triangulation of a set of points $R \subseteq P$ in the plane. We will use Ξ when R is clear from the context. For any triangle $\Delta \in \Xi$, let D_Δ be the Delaunay disk of Δ, and let P_Δ be the set of points of P contained in D_Δ. Similarly, for any edge $e \in \Xi$, let Δ_e^1 and Δ_e^2 be the two triangles in Ξ adjacent to e, and $P_e = P_{\Delta_e^1} \bigcup P_{\Delta_e^2}$. If e is on the convex-hull, then one of the triangles is taken to be the halfplane defined by e not containing R.

In order to prove that the algorithm gives the desired result, the following theorems regarding the size of an ϵ-net will be useful. Let $f(\epsilon)$ be the size of the smallest ϵ-net for any set P of points in $\mathbb{R}^2$ under disk ranges.

Lemma 2.1 ([4])**.** *For $\frac{2}{3} < \epsilon < 1$, $f(\epsilon) \leq 2$, and can be computed in $O(n \log n)$ time.*

Lemma 2.2. *For $\frac{1}{2} < \epsilon \leq \frac{2}{3}$, $f(\epsilon) \leq 10$ and can be computed in $O(n \log n)$ time.*

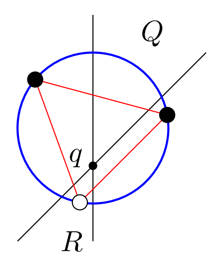


Figure 1: Setup around q.

Proof. Divide the plane into 4 quadrants with 2 lines, intersecting at a point q, such that each quadrant contains $n/4$ points. Using the Ham-Sandwich theorem, this can be done in linear time [26]. Create a $\frac{2}{3}$-net for each quadrant, using Lemma 2.1. Add these 8 points to the ϵ-net of P. If $q \in P$ then add q to the ϵ-net; otherwise let Δ be the triangle in the Delaunay triangulation of P that contains the point q. Add the two vertices of Δ that are in the opposite quadrants to the ϵ-net. The resulting size of the net is at most 10. Denote the quadrant without a vertex of the Delaunay triangle inside it by Q and its opposite quadrant by R. If a disk D intersects at most 3 quadrants and does not contain any of the points from the $\frac{2}{3}$-net in each of those quadrants, it can contain only at most $3 \cdot \frac{2}{3} \cdot \frac{n}{4} = \frac{n}{2}$ points. On the other hand, if D contains points from each of the 4 quadrants, then it must contain points from Q and R that are outside of the Delaunay disk D_Δ of Δ (as D_Δ is empty of points of

Algorithm 1: Compute ϵ-nets

Data: Compute ϵ-net, given P: set of n points in $\mathbb{R}^2$, $\epsilon > 0$ and c_1.

if $\epsilon n < 13$ **then**
  Return P

Pick each point $p \in P$ into R independently with probability $\frac{c_1}{\epsilon n}$.
if $|R| \leq c_1/2\epsilon$ *or* $|R| \geq 2c_1/\epsilon$ **then**
  restart algorithm.

Compute the Delaunay triangulation Ξ of R.
for *triangles* $\Delta \in \Xi$ **do**
  Compute the set of points $P_\Delta \subseteq P$ in Delaunay disk D_Δ of Δ.

for *edges* $e \in \Xi$ **do**
  Let Δ_e^1 and Δ_e^2 be the two triangles adjacent to e, $P_e = P_{\Delta_e^1} \cup P_{\Delta_e^2}$.
  Let $\epsilon' = (\frac{\epsilon n}{|P_e|})$ and compute a ϵ'-net R_e for P_e depending on the cases below:
  if $\frac{2}{3} < \epsilon' < 1$ **then**
    compute using Lemma 2.1.
  if $\frac{1}{2} < \epsilon' \leq \frac{2}{3}$ **then**
    compute using Lemma 2.2.
  if $\epsilon' \leq \frac{1}{2}$ **then**
    compute recursively.

Return $(\bigcup_e R_e) \cup R$.

P). Then if D does not contain any of the two vertices of Δ in the opposite quadrants (already added to the ϵ-net), it must pierce D_Δ (i.e., intersect four times), a contradiction. □

Call a tuple $(\{p, q\}, \{r, s\})$, where $p, q, r, s \in P$, a ***Delaunay quadruple*** if $int(\Delta(pqr)) \cap int(\Delta(pqs)) = \emptyset$. Define its *weight*, denoted $W_{(\{p,q\},\{r,s\})}$, to be the number of points of P in $D_{pqr} \cup D_{pqs}$. Let $\mathcal{T}_{\leq k}$ be a set of Delaunay quadruples of P of weight at most k and similarly $\mathcal{T}_k$ denotes the set of Delaunay quadruples of weight exactly k. Similarly, a ***Delaunay triple*** is given by $(\{p, q\}, \{r\})$, where $p, q, r \in P$. Define its *weight*, denoted $W_{(\{p,q\},\{r\})}$, to be the number of points of P in $D_{pqr} \cup D_{pq\bar{r}}$. Let $\mathcal{S}_{\leq k}$ be a set of Delaunay triples of P of weight at most k, and $\mathcal{S}_k$ denotes the set of Delaunay triples of weight exactly k.

One can upper bound the size of $\mathcal{T}_{\leq k}$, $\mathcal{S}_{\leq k}$ and using it, we derive an upper bound on the expected number of sub-problems with a certain number of points.

Claim 2.3. $|\mathcal{T}_{\leq k}| \leq (e^3/9)nk^3$ *asymptotically and* $|\mathcal{T}_{\leq k}| \leq (3.1)nk^3$ *for* $k \geq 13$.

Proof. The proof is an application of the Clarkson-Shor technique [26]. Pick each point in P independently with probability p_{cs} to get a random sample R_{cs}. Count the expected number of edges in the Delaunay triangulation of R_{cs} in two ways. On one hand, it is simply less than $3E[|R_{cs}|] = 3np_{cs}$. On the other hand, it is:

$$\begin{aligned}
3np_{cs} &\geq \mathbf{E}[\text{Number of Delaunay edges in } R_{cs}] = \sum_{p,q\in P} \Pr[\{p,q\} \text{ is a Delaunay edge of } R_{cs}] \\
&\geq \sum_{p,q\in P}\sum_{r,s\in P} \Pr[(D_{pqr}\cup D_{pqs})\cap R_{cs}=\emptyset] \qquad \text{(disjoint events)} \\
&\geq \sum_{(\{p,q\},\{r,s\})\in\mathcal{T}_{\leq k}} \Pr[(D_{pqr}\cup D_{pqs})\cap R_{cs}=\emptyset] \\
&\geq \sum_{(\{p,q\},\{r,s\})\in\mathcal{T}_{\leq k}} p_{cs}^4\cdot(1-p_{cs})^k = |\mathcal{T}_{\leq k}|\cdot p_{cs}^4\cdot(1-p_{cs})^k
\end{aligned}$$

Therefore $|\mathcal{T}_{\leq k}| \leq 3np_{cs}/(p_{cs}^4(1-p_{cs})^k)$ and a simple calculation gives that setting $p_{cs} = \frac{3}{k+3}$ minimizes the right hand side. Then $|\mathcal{T}_{\leq k}| \leq 3n\frac{3}{k+3}/((\frac{3}{k+3})^4(1-\frac{3}{k+3})^k) = nk^3\frac{1}{9}(1+\frac{3}{k})^{k+3}$, and the claim follows. ☐

Claim 2.4. $|\mathcal{S}_{\leq k}| \leq (e^2/4)nk^2$ *asymptotically and* $|\mathcal{S}_{\leq k}| \leq (2.14)nk^2$ *for* $k \geq 13$.

Proof. Pick each point in P independently with probability p_{cs} to get a random sample R_{cs}. Count the expected number of edges in the Delaunay triangulation of R_{cs} that lie on the boundary of the Delaunay triangulation, i.e., adjacent to exactly one triangle, in two ways. On one hand, it is exactly the number of edges in the convex-hull of R_{cs}, therefore at most $E[|R_{cs}|] = np_{cs}$. Counted another way, it is:

$$\begin{aligned}
np_{cs} &\geq \mathbf{E}[\text{Number of boundary Delaunay edges in } R_{cs}] = \sum_{p,q\in P} \Pr[\{p,q\} \text{ is a boundary Delaunay edge of } R_{cs}] \\
&\geq \sum_{p,q\in P}\sum_{r\in P} \Pr[(D_{pqr}\cup D_{pq\bar{r}})\cap R_{cs}=\emptyset] \qquad \text{(disjoint events)} \\
&\geq \sum_{(\{p,q\},\{r\})\in\mathcal{S}_{\leq k}} \Pr[(D_{pqr}\cup D_{pq\bar{r}})\cap R_{cs}=\emptyset] \\
&\geq \sum_{(\{p,q\},\{r\})\in\mathcal{S}_{\leq k}} p_{cs}^3\cdot(1-p_{cs})^k = |\mathcal{S}_{\leq k}|\cdot p_{cs}^3\cdot(1-p_{cs})^k
\end{aligned}$$

Setting $p_{cs} = \frac{2}{k+2}$ gives the required result. ☐

Claim 2.5.

$$\boldsymbol{E}\Big[\big|\{e\in\Xi \ \mid\ k_1\epsilon n \leq |P_e| \leq k_2\epsilon n\}\big|\Big] \leq \frac{(3.1)c_1^3}{\epsilon e^{k_1c_1}}(k_1^3c_1 + 3.7k_2^2) \ \textit{ if } \ \epsilon n \geq 13.$$

Proof. The crucial observation is that two points $\{p,q\}$ form an edge in Ξ with two adjacent triangles $\Delta(pqr), \Delta(pqs) \in \Xi$ **iff** $\{p,q,r,s\} \subseteq R$ and none of the points of P in $D_{pqr}\cup D_{pqs}$ are picked in R (i.e, the points p,q,r,s form the Delaunay tuple $(\{p,q\},\{r,s\})$). Or $\{p,q\}$ form an edge on the convex-hull of Ξ with one adjacent triangle $\Delta(pqr)$ **iff** $\{p,q,r\} \subseteq R$ and none of the points of P in $D_{pqr}\cup D_{pq\bar{r}}$ are picked in R.

Let $\chi_{(\{p,q\},\{r,s\})}$ be the random variable that is 1 iff $\{p,q\}$ form an edge in Ξ and their two adjacent triangles are $\Delta(pqr)$ and $\Delta(pqs)$. Let $\chi_{(\{p,q\},\{r\})}$ be the random variable that is 1 iff $\{p,q\}$ form an edge in Ξ

with exactly one adjacent triangle $\Delta(pqr)$. Noting that every edge in Ξ must come from either a Delaunay quadruple or a Delaunay triple,

$$\begin{aligned}\mathbf{E}[|\{e \mid k_1\epsilon n \le |P_e| \le k_2\epsilon n\}|] \;=\; & \sum_{\substack{p,q,r,s\in P\\ k_1\epsilon n \le W_{(\{p,q\},\{r,s\})} \le k_2\epsilon n}} \Pr[\chi_{(\{p,q\},\{r,s\})}=1] \;+ \\ & \sum_{\substack{p,q,r\in P\\ k_1\epsilon n \le W_{(\{p,q\},\{r\})} \le k_2\epsilon n}} \Pr[\chi_{(\{p,q\},\{r\})}=1]\end{aligned}$$

The second term is asymptotically smaller, so we bound it somewhat loosely:

$$\begin{aligned}\sum_{\substack{p,q,r\in P\\ k_1\epsilon n \le W_{(\{p,q\},\{r\})} \le k_2\epsilon n}} \Pr[\chi_{(\{p,q\},\{r\})}=1] \;\le\; & \sum_{\substack{p,q,r\\ k_1\epsilon n \le W_{(\{p,q\},\{r\})} \le k_2\epsilon n}} (c_1/\epsilon n)^3(1-c_1/\epsilon n)^{W_{(\{p,q\},\{r\})}} \\ \le\; & |\mathcal{S}_{\le k_2\epsilon n}| \cdot (c_1/\epsilon n)^3(1-c_1/\epsilon n)^{k_1\epsilon n} \\ \le\; & (2.14)n(k_2\epsilon n)^2 \cdot (c_1/\epsilon n)^3 \cdot e^{-c_1k_1} = \frac{(2.14)k_2^2c_1^3}{\epsilon e^{c_1k_1}}.\end{aligned}$$

Now we carefully bound the first term:

$$\begin{aligned}\sum_{\substack{p,q,r,s\in P\\ k_1\epsilon n \le W_{(\{p,q\},\{r,s\})} \le k_2\epsilon n}} \Pr[\chi_{(\{p,q\},\{r,s\})}=1] \;\le\; & \sum_{i=k_1\epsilon n}^{k_2\epsilon n} \sum_{\substack{p,q,r,s\\ W_{(\{p,q\},\{r,s\})}=i}} \Pr[\chi_{(\{p,q\},\{r,s\})}=1] \\ \le\; & \sum_{i=k_1\epsilon n}^{k_2\epsilon n} \sum_{\substack{p,q,r,s\\ W_{(\{p,q\},\{r,s\})}=i}} (c_1/\epsilon n)^4(1-c_1/\epsilon n)^i \\ \le\; & \sum_{i=k_1\epsilon n}^{k_2\epsilon n} |\mathcal{T}_i|(c_1/\epsilon n)^4(1-c_1/\epsilon n)^i\end{aligned}$$

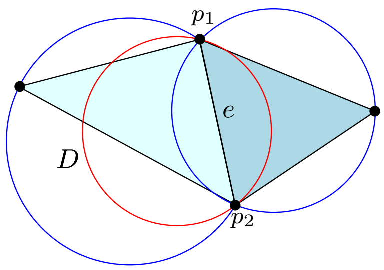
As the above summation is exponentially decreasing as a function of i, it is maximized when $|\mathcal{T}_{i_0}| = \max|\mathcal{T}_{\le i_0}|$ where $i_0 = k_1\epsilon n$, and $|\mathcal{T}_i| = \max|\mathcal{T}_{\le i}| - \max|\mathcal{T}_{\le i-1}|$ and so on. Using Claim 2.3 we obtain:

$$\begin{aligned}\le\; & |\mathcal{T}_{\le k_1\epsilon n}| \cdot (c_1/\epsilon n)^4(1-c_1/\epsilon n)^{k_1\epsilon n} + \sum_{i=k_1\epsilon n+1}^{k_2\epsilon n} (|\mathcal{T}_{\le i}| - |\mathcal{T}_{\le i-1}|) \cdot (c_1/\epsilon n)^4(1-c_1/\epsilon n)^i \\ \le\; & (3.1)n(k_1\epsilon n)^3 \cdot (c_1/\epsilon n)^4(1-c_1/\epsilon n)^{k_1\epsilon n} + \sum_{i=k_1\epsilon n+1}^{k_2\epsilon n} (3.1)n \cdot 3i^2 \cdot (c_1/\epsilon n)^4(1-c_1/\epsilon n)^i \\ \le\; & (3.1)\frac{k_1^3c_1^4e^{-k_1c_1}}{\epsilon} + (3.1)\frac{3k_2^2c_1^4}{\epsilon^2 n}\sum_{i=k_1\epsilon n+1}^{k_2\epsilon n}(1-c_1/\epsilon n)^i \\ \le\; & (3.1)\frac{k_1^3c_1^4e^{-k_1c_1}}{\epsilon} + (3.1)\frac{3k_2^2c_1^4}{\epsilon^2 n}\frac{(1-c_1/\epsilon n)^{k_1\epsilon n}}{c_1/\epsilon n} \le \frac{(3.1)c_1^3}{\epsilon e^{k_1c_1}}(k_1^3c_1 + 3k_2^2).\end{aligned}$$

The proof follows by summing up the two terms. $\square$

Using the above facts we can prove the main result.

Lemma 2.6. *Algorithm* COMPUTE ϵ-NET *computes an ϵ-net of expected size* $13.4/\epsilon$.



Proof. First we show that the algorithm computes an ϵ-net. Take any disk D with center c containing ϵn points of P, and not hit by the initial random sample R. Increase its radius while keeping its center c fixed until it passes through a point, say p_1 of R. Now further expand the disk by moving c in the direction $\vec{p_1 c}$ until its boundary passes through a second point p_2 of R. The edge e defined by p_1 and p_2 belongs to Ξ, and the two extreme disks in the pencil of empty disks through p_1 and p_2 are the disks $D_{\Delta_e^1}$ and $D_{\Delta_e^2}$. Their union covers D, and so D contains ϵn points out of the set P_e. Then the net R_e computed for P_e must hit D, as $\epsilon n = (\epsilon n/|P_e|) \cdot |P_e|$.

For the expected size, clearly, if $\epsilon n < 13$ then the returned set is an ϵ-net of size $\frac{13}{\epsilon}$. Note that no recursive call of the algorithm will terminate due to this condition since the number of points a disk has to contain to be hit remains unchanged. Otherwise we can calculate the expected number of points added to the ϵ-net during solving the sub-problems . We simply group them by the number of points in them. Set $E_i = \{e \mid 2^i \epsilon n \leq |P_e| < 2^{i+1}\epsilon n\}$ for $i \geq 1$, and let us denote the size of the ϵ-net returned by our algorithm with $f'(\epsilon)$. Note that $|P_e| < 2\epsilon n$ is handled by Lemma 2.1 and 2.2. Then

$$\begin{aligned}
\mathbf{E}\left[f'(\epsilon)\right] = \mathbf{E}[|R|] + \mathbf{E}\Big[|\bigcup_{e\in\Xi} R_e|\Big] \quad &= \quad \frac{c_1}{\epsilon} + \mathbf{E}[|\{e \mid \epsilon n \leq |P_e| < 3\epsilon n/2\}|] \cdot f(2/3) \\
&\quad + \mathbf{E}[|\{e \mid 3\epsilon n/2 \leq |P_e| < 2\epsilon n\}|] \cdot f(1/2) \\
&\quad + \sum_{i=1} \mathbf{E}\left[\sum_{e\in E_i} f'\left(\frac{\epsilon n}{|P_e|}\right)\right]
\end{aligned}$$

Noting that $\mathbf{E}[\sum_{e\in E_i} f'(\frac{\epsilon n}{|P_e|}) \mid |E_i| = t] \leq t\mathbf{E}[f'(1/2^{i+1})]$, we get

$$\mathbf{E}\left[\sum_{e\in E_i} f'\left(\frac{\epsilon n}{|P_e|}\right)\right] = \mathbf{E}\left[\mathbf{E}[\sum_{e\in E_i} f'\left(\frac{\epsilon n}{|P_e|}\right) |E_i]\right] \leq \mathbf{E}\left[|E_i| \cdot \mathbf{E}[f'(1/2^{i+1})]\right] = \mathbf{E}[|E_i|] \cdot \mathbf{E}[f'(1/2^{i+1})]$$

as $|E_i|$ and $f'(\cdot)$ are independent. As $\epsilon' = \frac{\epsilon n}{|P_e|} > \epsilon$, by induction, assume $\mathbf{E}[f'(\epsilon')] \leq \frac{13.4}{\epsilon'}$. Then

$$\begin{aligned}
\mathbf{E}\left[f'(\epsilon)\right] \quad &\leq \quad \frac{c_1}{\epsilon} + \frac{(3.1)\cdot c_1^3(c_1 + 8.34)}{\epsilon e^{c_1}} \cdot 2 + \frac{(3.1)\cdot c_1^3((3/2)^3 c_1 + 14.8)}{\epsilon e^{3c_1/2}} \cdot 10 \\
&\quad + \sum_{i=1} \frac{(3.1)\cdot c_1^3(2^{3i}c_1 + 3.7\cdot 2^{2i+2})}{\epsilon e^{c_1 2^i}} \cdot 13.4 \cdot 2^{i+1} \\
&\leq \frac{12}{\epsilon} + \frac{1.339}{\epsilon} + \frac{0.045}{\epsilon} + \frac{0.002}{\epsilon} \leq \frac{13.4}{\epsilon}
\end{aligned}$$

by setting $c_1 = 12$ which is close to the optimal solution of the recurrence relation. □

Finally, we bound the expected running time of the algorithm.

Lemma 2.7. *Algorithm* COMPUTE ϵ-NET *runs in expected time* $O(n \log n)$.

Proof. Note that $\mathbf{E}[|R|] = c_1/\epsilon$. First we bound the expected total size of all the sets P_e:

$$\begin{aligned}
\mathbf{E}\Big[\sum_{e\in\Xi} |P_e|\Big] \quad &\leq \quad \mathbf{E}[|\{e \mid 0 \leq |P_e| < \epsilon n\}|] \cdot \epsilon n + \sum_{i=0} \mathbf{E}[|\{e \mid 2^i\epsilon n \leq |P_e| < 2^{i+1}\epsilon n\}|] \cdot 2^{i+1}\epsilon n \\
&\leq \quad O(\frac{\epsilon n}{\epsilon}) + \sum_{i=0} O\left(\frac{(2^i)^3}{\epsilon e^{2^i c_1}}\right) \cdot 2^{i+1}\epsilon n = O(n),
\end{aligned}$$

as the last summation is a geometric series. This implies that the expected total number of incidences between points in P, and Delaunay disks in Ξ is $O(n)$. The Delaunay triangulation of R can be computed in expected time $O(1/\epsilon \log 1/\epsilon)$. Steps 7-8 compute, for each Delaunay disk $D \in \Xi$, the list of points contained in D. This can be computed in $O(n \log 1/\epsilon)$ time by instead finding, for each $p \in P$, the list of Delaunay disks in Ξ containing p, as follows. First do point-location in Ξ to locate the triangle Δ containing p, in expected time $O(\log 1/\epsilon)$. Clearly D_Δ contains p. Now starting from Δ, do a breadth-first search in the dual planar graph of the Delaunay triangulation to find the maximally connected subset of triangles (vertices in the dual graph) whose Delaunay disks contain p. As each vertex in the dual graph has degree at most 3, this takes time proportional to the discovered list of triangles, which as shown earlier is $O(n)$ over all $p \in P$. The correctness follows from the following:

Fact 2.8. *Given a Delaunay triangulation Ξ on R and any point $p \in \mathbb{R}^2$, the set of triangles in Ξ whose Delaunay disks contain p form a connected sub-graph in the dual graph to Ξ.*

Proof. This can be seen by lifting P to $\mathbb{R}^3$ via the Veronese mapping, where it follows from the fact that the faces of a convex polyhedron that are visible from any exterior point are connected. □

Note that the probability of restarting the algorithm (lines 4-5) at any call is at most a constant. Therefore it is re-started expected at most a constant number of times, and so the expected running time, denoted by $T(n)$:

$$\mathbf{E}[T(n)] \;=\; O(1/\epsilon \log 1/\epsilon) + O(n \log 1/\epsilon) + \sum_{e \in \Xi} \mathbf{E}[T(|P_e|)] \leq O(n \log 1/\epsilon) + \sum_{e \in \Xi} \mathbf{E}[T(|P_e|)]$$

Similarly to previous calculations we have that

$$\begin{aligned}
\mathbf{E}[T(n)] \;\leq\; & O(n \log 1/\epsilon) + \frac{(3.1) \cdot c_1^3 (c_1 + 8.34)}{\epsilon e^{c_1}} \cdot O(3\epsilon n/2 \log(3\epsilon n/2)) \\
& + \frac{(3.1) \cdot c_1^3 ((3/2)^3 c_1 + 14.8)}{\epsilon e^{3c_1/2}} \cdot O(2\epsilon n \log(2\epsilon n)) \\
& + \sum_{i=1} \frac{(3.1) \cdot c_1^3 (2^{3i} c_1 + 3.7 \cdot 2^{2i+2})}{\epsilon e^{c_1 2^i}} \cdot \mathbf{E}[T(2^{i+1}\epsilon n)] \\
& \leq dn \log n + \sum_{i=1} \frac{(3.1) \cdot c_1^3 (2^{3i} c_1 + 3.7 \cdot 2^{2i+2})}{\epsilon e^{c_1 2^i}} \cdot \mathbf{E}[T(2^{i+1}\epsilon n)]
\end{aligned}$$

for a constant d coming from the constants above, as well as in Delaunay triangulation, point-location and list-construction computations. Setting $\mathbf{E}[T(k)] = ck \log k$ satisfies the above inequality for $c \geq 2d$, since

$$\begin{aligned}
\mathbf{E}[T(n)] \;\leq\; & dn \log n + \sum_{i=1} \frac{(3.1) \cdot c_1^3 (2^{3i} c_1 + 3.7 \cdot 2^{2i+2})}{\epsilon e^{c_1 2^i}} \cdot c(2^{i+1}\epsilon n) \log(2^{i+1}\epsilon n) \\
\leq\; & dn \log n + (cn \log n) \sum_{i=1} \frac{2^{i+1} \cdot (3.1) \cdot 12^3 (2^{3i} \cdot 12 + 3.7 \cdot 2^{2i+2})}{e^{12 \cdot 2^i}} \\
\leq\; & dn \log n + cn \log n \cdot \frac{1}{2} \leq cn \log n, \text{ for } \; c \geq 2d.
\end{aligned}$$

□

3 Implementation and Experiments

In this section we present experimental results for our algorithm running on a machine equipped with an Intel Core i7 870 processor with 4 cores each running at 2.93 GHz and with 16 GB main memory. All our implementations are single threaded in order to have a fair comparison. For nearest-neighbors and Delaunay triangulations, we use the well-known geometry library CGAL. It computes Delaunay triangulations in expected $O(n \log n)$ time. Instead of computing centerpoints, we will recurse for all values of ϵ'; this results in simple efficient code, at the cost of slightly larger constants.

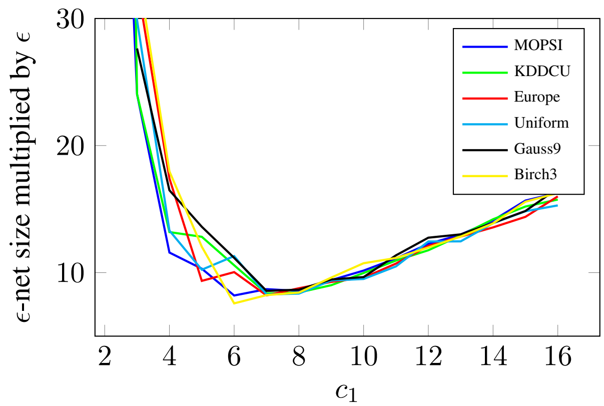


Figure 2: ϵ-net size multiplied by ϵ for 4 sets, $\epsilon = 0.01$.

In order to empirically validate the size of the ϵ-net obtained by our random sampling algorithm we have utilized several datasets in [1]. The *MOPSI Finland* dataset contains $13,467$ locations of users in Finland. The *KDDCUP04Bio* dataset contains the first 2 dimensions of a protein dataset with $145,751$ entries. The *Europe* and *Birch3* datasets have $169,308$ and $100,000$ entries respectively. We have created two random data sets *Uniform* and *Gauss9* with $50,000$ and $90,000$ points. The former is sampled from a uniform distribution while the latter is sampled from 9 different gaussian distributions whose means and covariance matrices are randomly generated. Setting the probability for random sampling to $\frac{12}{\epsilon \cdot n}$ results in approximately $\frac{12}{\epsilon}$ sized nets for nearly all datasets, as expected by our analysis. We note however, that in practice setting c_1 to 7 gives smaller size ϵ-nets, of size around $\frac{9}{\epsilon}$ (while our analysis would give an upper bound of $\frac{82}{\epsilon}$ in this case). See Figure 2 for the dependency of the net size on c_1 while setting ϵ to 0.01. In Table 1 we list ϵ-net sizes for different values of ϵ while setting c_1 to 12.

Dataset	ϵ-net size			
	$\epsilon = 0.2$	$\epsilon = 0.1$	$\epsilon = 0.01$	$\epsilon = 0.001$
MOPSI Finland	83	128	1226	12011
KDDCUP04Bio	55	118	1176	11902
Europe	69	119	1205	12043
Birch 3	58	125	1198	11878
Uniform	70	109	1245	12034
Gauss9	58	120	1275	12011

Table 1: ϵ-net sizes for various point sets, $c_1 = 12$.

4 Conclusion

In this paper we have improved upon the constants in the previous construction of ϵ-nets for disks in the plane. Our method gives an efficient practical algorithm for computing such ϵ-nets, which we have implemented and tested on a variety of data-sets. We conclude with a list of open problems:

- Currently the best known lower-bound for ϵ-nets for disks in the plane is the same as the $2/\epsilon$ bound for halfplanes in $\mathbb{R}^2$. It remains an interesting question to improve this lower-bound, or improve the upper-bounds given in this paper.
- The algorithm of Agarwal and Pan [2] uses a number of heavy tools (dynamic range reporting, dynamic approximate range counting) that hinders an efficient and practical implementation of their algorithm. Some partial progress has been made recently [7], but much work needs to be done to derive a more practical method with provable guarantees.

Acknowledgments. The authors would like to thank the reviewers for comments and suggestions that substantially improved the content and style of the paper.

References

[1] Clustering datasets, http://cs.joensuu.fi/sipu/datasets/.

[2] Pankaj K. Agarwal and Jiangwei Pan. Near-linear algorithms for geometric hitting sets and set covers. In *Symposium on Computational Geometry*, pages 271–279, 2014.

[3] Christoph Ambühl, Thomas Erlebach, Matús Mihalák, and Marc Nunkesser. Constant-factor approximation for minimum-weight (connected) dominating sets in unit disk graphs. In *APPROX-RANDOM*, pages 3–14, 2006.

[4] Pradeesha Ashok, Umair Azmi, and Sathish Govindarajan. Small strong epsilon nets. *Comput. Geom.*, 47(9):899–909, 2014.

[5] Herve Bronnimann and Michael Goodrich. Almost optimal set covers in finite VC-dimension. *Discrete & Computational Geometry*, 14(4):463–479, 1995.

[6] Norbert Bus, Shashwat Garg, Nabil H. Mustafa, and Saurabh Ray. Improved local search for geometric hitting set. In *32nd International Symposium on Theoretical Aspects of Computer Science*, pages 184–196, 2015.

[7] Norbert Bus, Nabil H. Mustafa, and Saurabh Ray. Geometric hitting sets for disks: Theory and practice. In *Proceedings of the 23rd Annual European Symposium on Algorithms*, pages 903–914, 2015.

[8] Gruia Călinescu, Ion I. Mandoiu, Peng-Jun Wan, and Alexander Zelikovsky. Selecting forwarding neighbors in wireless ad hoc networks. *MONET*, 9(2):101–111, 2004.

[9] Paz Carmi, Matthew J. Katz, and Nissan Lev-Tov. Covering points by unit disks of fixed location. In *ISAAC*, pages 644–655, 2007.

[10] Timothy M. Chan, Elyot Grant, Jochen Könemann, and Malcolm Sharpe. Weighted capacitated, priority, and geometric set cover via improved quasi-uniform sampling. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1576–1585, 2012.

[11] Bernard Chazelle and Joel Friedman. A deterministic view of random sampling and its use in geometry. *Combinatorica*, 10(3):229–249, 1990.

[12] Kenneth Clarkson and Kasturi Varadarajan. Improved approximation algorithms for geometric set cover. In *Symposium on Computational Geometry*, pages 135–141, 2005.

[13] Francisco Claude, Gautam K. Das, Reza Dorrigiv, Stephane Durocher, Robert Fraser, Alejandro López-Ortiz, Bradford G. Nickerson, and Alejandro Salinger. An improved line-separable algorithm for discrete unit disk cover. *Discrete Math., Alg. and Appl.*, 2(1):77–88, 2010.

[14] Gautam K. Das, Robert Fraser, Alejandro Lopez-Ortiz, and Bradford G. Nickerson. On the discrete unit disk cover problem. *International Journal on Computational Geometry and Applications*, 22(5):407–419, 2012.

[15] G. Even, D. Rawitz, and S. Shahar. Hitting sets when the VC-dimension is small. *Inf. Process. Lett.*, 95:358–362, 2005.

[16] Robert Fraser. *Algorithms for Geometric Covering and Piercing Problems*. PhD thesis, University of Waterloo, 2012.

[17] Shashidhara K. Ganjugunte. *Geometric Hitting Sets and Their Variants*. PhD thesis, Duke University, 2011.

[18] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, NY, 1979.

[19] Sariel Har-Peled, Haim Kaplan, Micha Sharir, and Shakhar Smorodinsky. Epsilon-nets for halfspaces revisited. *CoRR*, abs/1410.3154, 2014.

[20] D. Haussler and E. Welzl. Epsilon-nets and simplex range queries. *Discrete Comput. Geom.*, 2:127–151, 1987.

[21] D. S. Hochbaum and W. Maass. Fast approximation algorithms for a nonconvex covering problem. *J. Algorithms*, 8(3):305–323, 1987.

[22] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[23] János Komlós, János Pach, and Gerhard J. Woeginger. Almost tight bounds for epsilon-nets. *Discrete & Computational Geometry*, 7:163–173, 1992.

[24] Jirí Matousek. Efficient partition trees. *Discrete & Computational Geometry*, 8:315–334, 1992.

[25] Jirí Matousek. On constants for cuttings in the plane. *Discrete & Computational Geometry*, 20(4):427–448, 1998.

[26] Jirí Matousek. *Lectures in Discrete Geometry*. Springer-Verlag, New York, NY, 2002.

[27] Jirí Matousek, Raimund Seidel, and Emo Welzl. How to net a lot with little: Small epsilon-nets for disks and halfspaces. In *Proceedings of Symposium on Computational Geometry*, pages 16–22, 1990.

[28] Nabil H. Mustafa, Rajiv Raman, and Saurabh Ray. QPTAS for weighted geometric set cover on pseudodisks and halfspaces. *SIAM Journal on Computing*, 44(6):1650–1669, 2015.

[29] Nabil H. Mustafa and Saurabh Ray. Weak ϵ-nets have a basis of size $O(1/\epsilon \log 1/\epsilon)$. *Comp. Geom: Theory and Appl.*, 40(1):84–91, 2008.

[30] Nabil H. Mustafa and Saurabh Ray. Improved results on geometric hitting set problems. *Discrete & Computational Geometry*, 44(4):883–895, 2010.

[31] Nabil H. Mustafa and Saurabh Ray. Near-optimal generalisations of a theorem of Macbeath. In *31st International Symposium on Theoretical Aspects of Computer Science*, pages 578–589, 2014.

[32] János Pach and Gerhard J. Woeginger. Some new bounds for epsilon-nets. In *Symposium on Computational Geometry*, pages 10–15, 1990.

[33] Evangelia Pyrga and Saurabh Ray. New existence proofs epsilon-nets. In *Symposium on Computational Geometry*, pages 199–207, 2008.

[34] R. Raz and M. Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *Proceedings of STOC*, pages 475–484, 1997.

[35] Kasturi Varadarajan. Epsilon nets and union complexity. In *Proceedings of the 25th ACM Symposium on Computational Geometry, Aarhus, Denmark, June 8-10, 2009*, pages 11–16, 2009.

[36] Kasturi Varadarajan. Weighted geometric set cover via quasi-uniform sampling. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 641–648, 2010.