On Finding a Guard that Sees Most and a Shop that Sells Most

Otfried Cheong[1] Alon Efrat[2] Sariel Har-Peled[3]

October 3, 2003

Abstract

We present a near-quadratic time algorithm that computes a point inside a simple polygon P having approximately the largest visibility polygon inside P, and near-linear time algorithm for finding the point that will have approximately the largest Voronoi region when added to an n-point set. We apply the same technique to find the translation that approximately maximizes the area of intersection of two polygonal regions in near-quadratic time.

1 Introduction

We consider two problems where our goal is to find a point x such that the area of the region $V(x)$ "controlled" by x is as large as possible. In the first problem, we are given a simple polygon P, and $V(x)$ is the *visibility polygon* of x, that is, the region of points y inside P such that the segment xy does not intersect the boundary of P. In the second problem, we are given a set of points T, and $V(x)$ is the *Voronoi cell* of x in the Voronoi diagram of the set $T \cup \{x\}$, that is, the set of points that are closer to x than to any point in T.

In both problems, it is straightforward to write a closed formula describing the area of the region controlled by a point x. This area function (inside a region where $V(x)$ has the same combinatorial structure) is the sum of the areas of triangles that depend on the location of x. The function is therefore a sum of square roots of rational functions. This function is cumbersome to handle; in fact it is unknown whether comparing the value of such a function for a specific rational input to the square root of an integer number is in NP [OPP03]. It seems difficult to solve the problem analytically, and we resort to approximation.

In this paper we address the question of efficiently finding a point x that approximately maximizes the area of $V(x)$. More precisely, let $\mu(x)$ be the area of $V(x)$, and let $\mu_{opt} = \max_x \mu(x)$ be the area for the optimal solution. Given $\delta > 0$, we show how to find x_{app} such that $\mu(x_{app}) \geq (1-\delta)\mu_{opt}$.

The main motivation for our first problem arises from *art-gallery* or *sensor placement* problems. In a typical problem of this type, we are given a simple polygon P, and wish to find a set of points (*guards*) so that each point of P is seen by least one guard. This problem is NP-hard. Art-gallery problems have attracted a lot of research in the last thirty years [O'R87, Urr00]. A natural heuristic for solving art-gallery problems is to use a greedy approach based on area: We first find a guard that maximizes the area seen, next find a guard that sees the maximal area not seen by the first guard, and so on until each point of P is seen by some guard.

[1]Department of Mathematics and Computer Science, TU Eindhoven, P.O. Box 513, 5600 MB Eindhoven, The Netherlands; ocheong@win.tue.nl.

[2]Department of Computer Science, University of Arizona, Tucson, AZ 85721, USA; alon@cs.arizona.edu.

[3]Department of Computer Science, DCL 2111, University of Illinois, 1304 West Springfield Ave., Urbana, IL 61801, USA; http://www.uiuc.edu/~sariel; sariel@uiuc.edu. Work on this paper was partially supported by a NSF CAREER award CCR-0132901.

Ghosh [Gho87] used a similar greedy heuristic to obtain an $O(\log n)$-approximation on the number of guards needed to see an n-edge polygon, if guard locations are constrained to be on the vertices of P. (An improved algorithm obtains an $O(\log k_{opt})$-approximation [EH02].)

No approximation bounds for the greedy approach are known if guards can be located in the interior of P. However, for the related problem of maximizing the area seen by k guards, for a given number k, Hochbaum and Pathria [HP98] showed that k iterations of the greedy algorithm mentioned above construct a $(1-1/e)$-approximation to the more general set-cover problem. In Section 2.4, we show how to apply our result to the problem of finding k-guards that see as much as possible of the polygon P.

Ntafos and Tsoukalas [NT94] show how to find, for any $\delta > 0$, a guard that sees an area of size $(1-\delta)\mu_{opt}$. Their algorithm requires $O(n^5/\delta^2)$ time in the worst case. In Section 2.3, we give a probabilistic algorithm that finds such an approximation in time $O((n^2/\delta^4)\log^3(n/\delta))$ with high probability. We also show that approximating the largest visible polygon up to a constant factor is 3SUM-hard [GO95], implying that our algorithm is probably close to optimal as far as the dependency on n is concerned.

Our second problem is motivated by the task of placing a new supermarket such that it takes over as many customers as possible from the existing competition. If we assume that customers are uniformly distributed and shop at the nearest supermarket, then our task is indeed to find a point x such that the Voronoi region of x is as large as possible. The area of Voronoi regions has been considered before in the context of games, such as the Voronoi game [ACC$^+$01, CHLM02] or the Hotelling game [OBSC00]. As far as we know, the only previous paper discussing maximizing the Voronoi region of a new point is by Dehne et al. [DKS02], who show that the area function has only a single local maximum inside a region where the set of Voronoi neighbors does not change and is in convex position. They give an algorithm for finding (approximately) the optimal new point numerically based on Newton approximation.

In Section 3 we show that given a set T of n points and a $\delta > 0$, we can find a point x_{app} such that $\mu(x_{app}) \geq (1-\delta)\mu_{opt}$, where $\mu(x)$ is the area of the Voronoi region of x in the Voronoi diagram of $T \cup \{x\}$, and $\mu_{opt} = \max_x \mu(x)$. The (deterministic) running time of the algorithm is $O(n/\delta^2 + n\log n)$.

Our framework captures a variety of other problems, where the goal is to maximize the area of some region which depends on a multi-dimensional parameter. As an example of such a further application, we consider the problem of matching two planar shapes P and Q under translations. The area of overlap (or the area of the symmetric difference) of two planar regions is a natural measure of their similarity that is insensitive to noise [AFRW98, dBCD$^+$98]. Mount et al. [?] first studied the function mapping a translation vector to the area of overlap of a translated simple polygon P with another simple polygon Q, showing that it is continuous and piecewise polynomial of degree at most two. If n and m are the number of vertices of P and Q, respectively, then the function has $O((nm)^2)$ pieces, and can be computed within the same time bound. No algorithm is known that computes the translation maximizing the area of overlap that does not essentially construct the whole function graph. De Berg et al. [dBCD$^+$98] gave an $O((n+m)\log(n+m))$ time algorithm to solve the problem in the case of convex polygons, and gave a constant-factor approximation. Alt et al. [AFRW98] gave a constant-factor approximation for the minimum area of the symmetric difference of two convex polygons. Finally, de Berg et al. [dBGK$^+$03] consider the case where P is the union of n homothets of a convex body C, Q is the union of m homothets of C, the ratio between the largest and smallest homothet is bounded by a constant, and no point of the plane lies in more than a constant number of homothets. They compute a $(1-\varepsilon)$-approximation for the maximal area of overlap of P and Q in time $O(nm\log nm)$.

We consider the case where P and Q are polygonal regions of complexity n and m, where we assume $n \leq m$. In time $O(m\log n + n^2\log n)$ we can compute a translation maximizing the area of overlap up to ε times the area of Q—in other words, the error is *absolute* here. Note that this makes sense in this application: if the maximal area of overlap is only a small percentage of Q (say less than 1% of Q), then P and Q obviously do not match very well. In many applications it will be sufficient to know that there is no decent match possible. If a match of more than 1% is possible, then there is no difference between absolute and relative error. Finally, we consider the case of homothets studied by de Berg et al., and obtain an $O(m \log n + n \log^2 n)$ time algorithm computing a translation that maximizes the overlap, again up to an absolute error ε. This algorithms are probabilistic, and succeed with high probability.

Our algorithms are based on the theory of ε-approximations. Instead of measuring area directly, we estimate it by counting the number of points of an ε-approximation S inside the region. The estimate is sufficiently tight such that the point that maximizes it is a good approximation for the optimal solution. The beauty of this approach is that it turns a continuous problem into a discrete one: we only need to find the point x such that $V(x)$ contains the largest number of points of S. If, for $s \in S$, we define $W(s) = \{x \mid s \in V(x)\}$, then this problem can be solved by computing the arrangement of the $W(s)$, for $s \in S$, and inspecting each face of this arrangement.

In Section 2.1, we show how to apply this approach to our first problem, maximizing the visibility region. Unfortunately, it turns out that the size of the ε-approximation required is prohibitively large. This is mainly due to the fact that the area of the optimal solution might only be of the order of $1/n$, so our ε-approximation needs to guarantee an error of less than δ/n. In Sections 2.2 and 2.3 we show how to work around these problems, and improve the running time to near-quadratic.

In Section 3, we consider the Voronoi region problem. Here we can exploit the geometry of the optimal solution to decompose the problem into subproblems such that in each subproblem a small ε-approximation is sufficient.

We generate ε-approximations by random sampling. The reader may wonder why, if apparently random sampling works, we cannot simply generate a random sample X, compute $\mu(x)$ for each $x \in X$, and pick the sample point maximizing $\mu(x)$. Such an approach appears to work for maximizing the Voronoi region, but the required sample size would be prohibitively large. The approach doesn't work at all for maximizing the visibility region, as we will see in Section 2.1. Our indirect use of random sampling makes indeed all the difference.

2 Maximizing the visibility region

2.1 Using an ε-approximation

In the following, let P denote a simple polygon, $\mu(\cdot)$ denote the area measure, and assume that the area of P is 1; that is, $\mu(P) = 1$. Given a point $x \in P$, let $V_P(x)$ denote the *visibility polygon* of x inside P; that is, the region in P visible from x. Formally,

$$V_P(x) = \left\{ y \;\middle|\; xy \subseteq int(P) \right\}.$$

Note, that under this definition a visibility polygon is an open set. Let $\mu(x)$ denote the area of $V_P(x)$, and let $\mu_{opt} = \max_{x \in P} \mu(x)$ denote the maximal area.

Definition 2.1 For a set S of points in P, and a point $x \in P$, let

$$e_S(x) = \frac{|V_P(x) \cap S|}{|S|},$$

be the *estimate of the area* visible from x.

Consider the range space $(P, \mathcal{V})$, where $\mathcal{V} = \{V_P(x) \mid x \in P\}$. The set S is an ε*-approximation* for this range space if for any $x \in P$ we have (recall that $\mu(P) = 1$)

$$|e_S(x) - \mu(x)| \leq \varepsilon.$$

Valtr [Val98] showed that the VC-dimension of the range space $(P, \mathcal{V})$ is bounded by 23. By the ε-approximation theorem [AS00], a uniform random sample S of $O\big((d/\varepsilon^2) \log (d/\varepsilon\delta)\big)$ points from a range space of of VC dimension d is an ε-approximation for this range space with probability $\geq 1 - \delta$.

A uniform sample of points from P can be easily obtained by triangulating P, and first choosing the triangle (with probability proportional to its area), and then choosing a point from inside the triangle. Thus, this uniform sampling can be done in $O(\log n)$ expected time per sample point, after $O(n \log n)$ preprocessing (in fact, linear preprocessing is also achievable).

We note now that $\mu_{opt} \geq 1/(n-2)$, since this quantity is bounded from below by the area of the largest triangle in a triangulation of P. Let's assume that S is an ε-approximation for $\varepsilon = \delta/2n$, let $x_{app} \in P$ be the point maximizing $e_S(x_{app})$, and let $x_{opt} \in P$ be the point maximizing $\mu(x_{opt})$. Then we have

$$\mu(x_{app}) \geq e_S(x_{app}) - \delta/2n \geq e_S(x_{opt}) - \delta/2n \geq \mu(x_{opt}) - \delta/n \geq (1-\delta)\mu_{opt}.$$

In other words, the point $x_{app} \in P$ seeing the maximal number of points of S is a $(1-\delta)$-approximation to the point in P having the largest visibility polygon.

Now note that $s \in V_P(x)$ if and only if $x \in V_P(s)$. Let $\mathcal{W}_S = \left\{ V_P(s) \,\middle|\, s \in S \right\}$ be the set of visibility polygons defined by the points of S, and let $\mathcal{A}_P(S)$ denote the arrangement $\mathcal{A}(\mathcal{W}_S)$. Our problem has reduced to finding a point in P that is contained in the largest number of polygons in $\mathcal{W}_S$.

Lemma 2.2 *Given a simple polygon P, and a parameter $\delta > 0$, one can compute, in $O\big((n^5/\delta^4)\log^3(n/\delta)\big)$ time, a point $x \in P$, such that $\mu(x) \geq (1-\delta)\mu_{opt}$.*

Proof. Let $\varepsilon = \delta/2n$. A uniform random sample S of

$$M = O(1/\varepsilon^2 \log(1/\varepsilon)) = O\big((n^2/\delta^2)\log(n/\delta)\big)$$

points from P is an ε-approximation with high probability [AS00].

We compute, for each point $s \in S$, its visibility polygon $V_P(s)$ using sweeping. Let $\mathcal{W}_S$ be the resulting set of polygons. The complexity of the arrangement of $\mathcal{A}(\mathcal{W}_S)$ is $O(nM^2) = O\big((n^5/\delta^4)\log^2(n/\delta)\big)$; it can be computed in $O\big((n^5/\delta^4)\log^3(n/\delta)\big)$ time.

To see the bound on the complexity, observe that a segment inside P might intersect the boundary of a visibility polygon at most twice. The total number of edges of the visibility polygons in $\mathcal{W}_S$ is $O(nM)$, and each such segment contains at most $O(M)$ vertices of the arrangement, implying the bound stated. See [GMMN90] for details.

Finally, we perform a simple traversal of the arrangement, where we compute for each face the number of polygons of $\mathcal{W}_S$ that contain it. We pick a point in the face where this number is largest. ◻

The size of the sample used is too large to make the above algorithm attractive. There are two reasons for this: The value of ε has to be chosen sufficiently small to guarantee correctness for the extreme case where $\mu_{opt} \approx 1/n$. Furthermore, an ε-approximation is stronger than what is really required: it guarantees a good approximation for *any* range. In the next section we will see that testing a "small" (that is, polynomial) number of candidates is sufficient, and in Section 2.3 we will then deal with the problem of possibly small μ_{opt}.

At this point, the reader may wonder why we do not take a more direct approach of just sampling enough points, for each point computing its visibility polygon, and returning the largest visibility polygon computed. Somewhat surprisingly, this does not work, as demonstrated by the example depicted in Figure 1. Imagine that we stretch the horizontal and vertical corridors until each of them has area $1/n - 1/n^{10}$, while the central room has area $1/n^9$. With high probability, a random sample of size, say, $O(n)$ would have sample points only inside those corridors. Furthermore, the sample points would be "deep" in the corridors. As such, every random sample point would see an area $\leq 1/n$, while one can place a point in the central room that sees area $\geq 2/n$. In fact, the visibility arrangement we get for such a sample has quadratic complexity and no point is contained in more than two polygons.

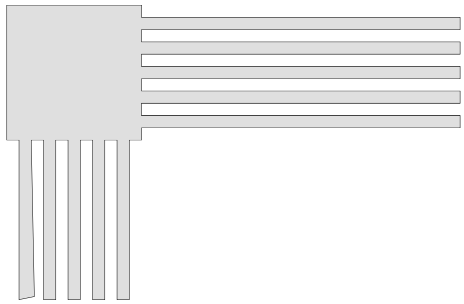

Figure 1: Canonical counterexample

2.2 Estimating the area directly

Given a point $x \in P$, we can estimate $\mu(x)$ by sampling a set S of points in P, and computing the fraction that is visible from x (we assume that $\mu(P) = 1$).

Lemma 2.3 *Let ν, δ be parameters, x be any point in P, such that $\mu(x) \geq \nu$ and $0 < \delta \leq 1$. Let S be a uniform sample from P of size $M \geq c_1 \frac{\log n}{\delta^2 \nu}$, where c_1 is an appropriate constant. Then,*

$$\mathbf{Pr}[|e_S(x) - \mu(x)| > \delta \cdot \mu(x)] \leq \frac{1}{n^{10}}.$$

Proof. This is immediate from the Chernoff inequality. Indeed, let $X_1, \ldots, X_M$ be indicator variables, such that $X_i = 1$ if and only if the ith sample point $s_i \in S$ is inside $V_P(x)$. Let $X = \sum_i X_i$, $c_1 = 44$ and

$$\rho = \mathbf{E}[X] = \mu(x)M \geq \nu \cdot c_1 \frac{\log n}{\delta^2 \nu} = 44 \frac{\log n}{\delta^2}.$$

By the (simplified form of the) Chernoff inequality [MR95], we have

$$\begin{aligned} \mathbf{Pr}\left[\left|\frac{|V_P(x) \cap S|}{|S|} - \mu(x)\right| > \delta \cdot \mu(x)\right] &= \mathbf{Pr}[|X - \rho| > \delta\rho] \\ &\leq e^{-\delta^2\rho/2} + e^{-\delta^2\rho/4} \leq 2\exp\left(-\frac{\delta^2\rho}{4}\right) \leq 2\exp(-11\log n) \leq \frac{1}{n^{10}}. \end{aligned}$$

□

The key observation in the above lemma is that because we are estimating $\mu(x)$ for a single fixed point x only, the sample we need is considerably smaller than the sample required by an ε-approximation, which guarantees the approximation bound for *every* point x at the same time. Naturally, we would like to use such a small sample in the algorithm of Lemma 2.2 and end up with a faster algorithm. However, one has now to be to careful, to argue that the random sample does not overestimate the area of the visibility polygon of some other point in the polygon. We do so by arguing that the random sample correctly estimate the visibility for all vertices of the visibility arrangement induced by S.

Lemma 2.4 *Let ν, δ be parameters, let S be a uniform sample from P of size $M \geq c_2 \frac{\log n}{\delta^2 \nu}$, where c_2 is an appropriate constant, and let x be a vertex of the arrangement $\mathcal{A}_P(S)$.*

1. *If $\mu(x) \leq \nu/4$, then $\mathbf{Pr}[|e_S(x)| \geq \nu/2] \leq \frac{1}{n^{10}}$.*

2. *If $\mu(x) \geq \nu/4$ then $\mathbf{Pr}[|e_S(x) - \mu(x)| > \delta \cdot \mu(x)] \leq \frac{1}{n^{10}}$.*

Proof. Observe, that x is the intersection of the boundary of two visibility polygons defined by two points of S. Let T be the set resulting from S by removing those two points. Clearly, the random sample T is independent of x. We have

$$e_T(x) = \frac{|V_P(x) \cap T|}{|T|} = \frac{|V_P(x) \cap S|}{|S|} \cdot \frac{|S|}{|T|} = e_S(x)\left(1 + \frac{2}{M-2}\right) \le e_S(x)\left(1 + \frac{\nu\delta^2}{10}\right),$$

since, by definition, the visibility polygons are open sets, and for c_2 large enough. In particular, we have $|e_S(x) - e_T(x)| \le \nu\delta^2/10$.

Thus, if $\mu(x) \le \nu/4$ then $e_S(x) \le e_T(x) \le \nu/2$ with high probability, by the Chernoff inequality. Alternatively, for $\mu(x) \ge \nu/4$, we have by Lemma 2.3 that

$$\mathbf{Pr}\left[|e_T(x) - \mu(x)| > \frac{\delta}{2} \cdot \mu(x)\right] \le \frac{1}{n^{10}}.$$

Observe that

$$|e_S(x) - \mu(x)| \le |e_T(x) - \mu(x)| + |e_S(x) - e_T(x)| \le |e_T(x) - \mu(x)| + \delta^2\nu/10,$$

and since $\mu(x) \ge \nu/4$, we have

$$\begin{aligned} \mathbf{Pr}[|e_S(x) - \mu(x)| > \delta\mu(x)] &\le \mathbf{Pr}\left[|e_T(x) - \mu(x)| + \frac{\delta^2\nu}{10} > \delta\mu(x)\right] \\ &\le \mathbf{Pr}\left[|e_T(x) - \mu(x)| > \frac{\delta}{2}\mu(x)\right] \le \frac{1}{n^{10}}. \end{aligned}$$

□

It is now natural to pick the vertex in the visibility arrangement contained in the largest number of visibility polygons as the best placement for a guard. The following lemma testifies that this indeed works with high probability.

Lemma 2.5 *Let ν, δ be parameters such that $\delta > 1/n^{0.1}$ and $\nu > 1/n$. Let S be a uniform sample from P of size $M \ge c_3 \frac{\log n}{\delta^2 \nu}$, where c_3 is an appropriate constant, and let x^* be the vertex of the arrangement $\mathcal{A}_P(S)$ that maximizes $e_S(x^*)$.*

1. *If $\mu_{opt} \le \nu/4$, then $\mathbf{Pr}[e_S(x^*) \ge \nu/2] \le \frac{1}{n^6}$.*
2. *If $\mu_{opt} \ge \nu/4$ then $\mathbf{Pr}[|e_S(x^*) - \mu_{opt}| > \delta \cdot \mu_{opt}(x^*)] \le \frac{1}{n^6}$.*

Proof. As we can argue quite similarly to Lemma 2.4, we only sketch the needed modifications. Consider the point x_{opt} that realizes $\max_{p \in P} \mu(p)$. Let x^* be a vertex of f, where f is the face of $\mathcal{A}_P(S)$ that contains x_{opt}. It is easy to verify, that $e_S(x^*)$ is "close" to $e_S(x_{opt})$. Thus, applying Lemma 2.4 to *all* vertices of $\mathcal{A}_P(S)$, we have that $e_S(x^*)$ is a good estimate to the maximum area visibility polygon in P. The arrangement $\mathcal{A}_P(S)$ has $O(nM^2) = o(n^4)$ vertices, by the assumptions on δ and ν. Thus, it follows that the probability that those estimates fail, is smaller than $(1/n^{10})o(n^4) < 1/n^6$. □

Lemma 2.5 yields an immediate algorithm for estimating the maximum area visibility polygon in P. Indeed, set $\nu = 1/n$, compute a sample S inside P of size $M = O((n \log n)/\delta^2)$, compute the arrangement $\mathcal{A}_P(S)$, and find the vertex that is contained in the largest number of visibility polygons induced by the points of S. Clearly, the overall running time of this algorithm is $O(nM^2 \log n) = O((n^3/\delta^4)\log^3(n/\delta))$. This algorithm succeeds with high probability. We conclude:

Theorem 2.6 *Given a simple polygon P, and a parameter $\delta > 0$, one can compute, in $O\big((n^3/\delta^4)\log^3(n/\delta)\big)$ time, a point $x \in P$, such that $\mu(x) \ge (1-\delta)\mu_{opt}$. This algorithm succeeds with high probability.*

2.3 Estimating the area of the optimal solution

The running time of Theorem 2.6 is dominated by the worst-case value of ν (which is a lower-bound on the area of the largest visibility polygon). Thus, it is natural to perform an exponential search for the right value of ν. Indeed, set $\nu = 1/2$, and use Lemma 2.5. Clearly, in near linear time, we either found the required visibility polygon, or alternatively, we know (with high probability) that $\mu_{opt} \leq 1/2$.

In the ith iteration, let $\nu_i = 1/2^i$, and we check whether $\mu_{opt} \leq \nu_i$, or alternatively, we get a $(1-\delta)$-approximation to the area of the largest visibility polygon.

What is the benefit in this approach? Well, in the ith iteration, we know that $\mu_{opt} \leq \nu_{i-1}$ (with high probability). Thus, with high probability, no point of $\mathcal{A}_P(S_i)$ is contained in more than $L = 2\nu_{i-1}M_i$ visibility polygons (see Lemma 2.12 below for a formal proof of this intuitive claim). Here S_i is the sample used in the ith iteration, of size $M_i = O\left(\frac{\log n}{\delta^2 \nu_i}\right)$. Furthermore,

$$L = 2\nu_{i-1}M_i = O\left(\frac{\log n}{\delta^2}\right).$$

The arrangement $\mathcal{A}_P(S_i)$ is *shallow*; that is, no point in this arrangement is contained in more than L visibility polygons.

Definition 2.7 A set S of points in P is *t-shallow* if no point in P is contained inside more than t visibility polygons of $\mathcal{W}_S$.

Lemma 2.8 *If S is t-shallow, then the complexity of the arrangement $\mathcal{A}_P(S)$ is $O(nkt)$, where n is the complexity of the polygon P, and $k = |S|$.*

Proof. The complexity of the union of k such visibility polygons is $O(nk)$ [GMMN90]. By Clarkson and Shor [CS89] this implies that the complexity of the at most t-level is $O(t^2 n(k/t)) = O(nkt)$. Since in our case, the at most t-level is the entire arrangement, the lemma follows. ∎

This implies that in the ith iteration, the algorithm computes an arrangement of complexity

$$O(nM_iL) = O\left(n\left(\frac{\log n}{\delta^2 \nu_i}\right)\frac{\log n}{\delta^2}\right).$$

Thus, the running time of the algorithm is dominated by the running time of the last iteration, which takes $O(nM_I L \log(nM_I L))$ time, where $I = \lceil \log_2 n \rceil$. We conclude:

Theorem 2.9 *Given a simple polygon P, and a parameter $\delta > 0$, one can compute, in $O\big((n^2/\delta^4)\log^3(n/\delta)\big)$ time, a point $x \in P$, such that $\mu(x) \geq (1-\delta)\mu_{opt}$. This algorithm succeeds with high probability.*

Interestingly, as pointed out to us by Jeff Erickson, this problem is 3SUM-hard [GO95], and as such the result of Theorem 2.9 is probably close to optimal, as this indicates that a subquadratic algorithm is high unlikely.

Lemma 2.10 *Given a simple polygon P, there is a constant $c > 0$, such the $(1-c)$-approximating the largest visible polygon in P is 3SUM-hard.*

Proof. The details of the proof are tedious but straightforward, and we only outline it. The basic idea is to carefully extend the example of Figure 1 for the case of arbitrary n lines.

Let L be a set of n lines with integer coefficients. It is well known that deciding whether L has three lines passing through a common point is 3SUM-hard. One can resize and translate L, in $O(n \log n)$ time, such that all the vertices of the arrangement of L lies in the unit square $[0.25, 0.75]^2$. Next, consider the axis parallel square S of side length M^{10} centered at the origin, and let us replace every line $\ell \in L$ by thickening it into a rectangle r_ℓ (i.e., take a Minkowski sum of ℓ with an appropriately small ball) such that the intersection of r_ℓ with S is of area 2, where $M \geq n^{10}$ is an appropriate large number which is function of the input. Furthermore, all those rectangles are disjoint outside the unit square (this can be guaranteed by picking M to be large enough). Let R denote the resulting set of rectangles. In fact, it is easy to guarantee by picking M large enough, that the topology of the union of the rectangles of R is identical to the topology of the union of lines (i.e., no faces outside the union disappear, etc).

Next, consider the polygon $P = (\cup_{r \in R} r) \cup [0,1]^2$. Clearly, we can compute P in $O(n \log n)$ time. Now, if there are three lines in L that pass through a common point, then there is a point that stabs three rectangles of R, and sees an area $\geq 3 \cdot 2 - o(1)$ inside P. Similarly, if there is no point that is contained in three lines of L, then clearly, every point inside P sees at most $2 \cdot 2 + 1 + o(1)$ area (i.e., area of two rectangles corresponding to two lines, and the area of the unit square, plus some minor additional portions of rectangles that might be locally visible).

This implies that there is a constant gap between the largest visible polygon in P depending on whether L has three lines that share a point. Thus, the problem of approximating the largest visible polygon up to a constant factor is 3SUM-hard. ∎

In many cases we do not expect to encounter inputs where the visibility polygon is truly small (that is $\approx 1/n$ of the total area of P). As such, the following corollary might be more useful.

Corollary 2.11 *Given a simple polygon P, and a parameter $\delta > 0$, one can compute, in $O\left(\frac{n}{\mu_{opt}\delta^4} \log^3 \frac{n}{\delta}\right)$ time, a point $x \in P$, such that $\mu(x) \geq (1-\delta)\mu_{opt}$. This algorithm succeeds with high probability.*

Lemma 2.12 *Let P be a simple polygon of area 1, such that the largest visible polygon in P of area $\leq \nu$, and let S be a uniform sample of size $M = \Omega((\log n)/\nu)$. Then, with high probability, no point in P sees more than $2M\nu$ points of S.*

Proof. Clearly, it is enough to prove this for all the vertices of the arrangement $\mathcal{A} = \mathcal{A}_P(S)$. Consider a vertex v of $\mathcal{A}$, defined by the visibility polygon of two points $p, q \in S$, and observe that the number of visibility polygons of $\mathcal{W}_S$ that covers v, is determined by the set $S \setminus \{p, q\}$, and as such it is a random variable independent of p, with expectation at most $\rho = \nu(M-2) = \Omega(\log n)$. As such, arguing as in Lemma 2.3, it follows by the Chernoff inequality, that the probability that p is contained in more than $2\nu M - 2$ polygons of $\mathcal{W}_S$ is smaller than $2\exp\left(-\frac{\rho}{4}\right)$. Namely, p is covered by at most $2M\nu - 2$ polygons of $\mathcal{W}_S$ with high probability. This implies the lemma, as the number of vertices of $\mathcal{W}_S$ is bounded by $O(nM^2)$. ∎

2.4 Finding a good set of guards

As discussed in the introduction, we want to use the greedy algorithm to find k "good" guards for P. Namely, at every step we pick a guard that sees as much as possible of the regions of the polygon of P not covered yet. We find the first guard using Theorem 2.9. To find the following guards, we need to slightly modify the algorithm, as the uncovered region is no longer a simple polygon. Indeed, assume that $\{g_1 \ldots g_i\}$ are guards that were already assigned, and let $Q_i \subseteq P$ be the region not seen by these guards, namely $Q_i = P \setminus \bigcup_i V_P(g_i)$. The complexity of Q_i is $O(ni)$, as Q_i is the complement of the union of i visibility polygons inside P [GMMN90]. We modify the algorithm to pick random sample points only from Q_i, and normalize the area of Q_i to be 1. It is straightforward to modify the algorithm of Theorem 2.6 to handle this more complicated case, and verify that the algorithm still work. The only major difference being that we set in Theorem 2.6, $\nu = 1/(ni)$ since Q_i can be decomposed into $O(ni)$ triangles. Hence the running time increases to $O\left((i^3 n^3/\delta^4) \log^3(n/\delta)\right)$. Similarly, the runtime of Theorem 2.9 increases to $O\left((i^2 n^2/\delta^4) \log^3(n/\delta)\right)$. To analyze the performance of this algorithm, we use the result of [HP98] that shows that a β-approximation algorithm to the heaviest set in a set-cover instance, when used repeatedly k times, results in a $1 - \exp(-\beta)$ approximation to the heaviest cover with k sets. Thus, plugging our approximation algorithm with the analysis of [HP98] yield the following result.

Theorem 2.13 *Given a simple polygon P, a parameter $\delta > 0$, and a positive integer k, one can compute, in $O\big((k^3 n^2/\delta^4)\log^3(n/\delta)\big)$ time, a set of k guards $\{g_1^* \dots g_k^*\} \subset P$ such that*

$$\mu\left(\bigcup_i V_P(g_i^*)\right) \geq (1 - \exp(\delta - 1)) \max_{\{g_1 \dots g_k\} \subset P} \mu\left(\bigcup_i V_P(g_i)\right)$$

This algorithm succeeds with high probability.

3 Maximizing the Voronoi region

Let T be a given fixed set of n points in the plane. For a point x not necessarily in T, let $V_T(x)$ denote the Voronoi region of x in the Voronoi diagram of $T \cup \{x\}$, and let $\mu(x)$ denote the area of $V_T(x)$. We are looking for a point x_{opt} maximizing $\mu_{opt} = \mu(x_{opt})$. For points x outside the convex hull of T, $\mu(x)$ would be infinite. There are quite a few ways of avoiding these boundary situations: using torus topology, restricting the point (i.e., supermarket) to lie within a polygon (i.e., city limits), or by adding a boundary that acts as an additional site. In the following we choose the first option, and assume the input is a set of points in a unit square with torus topology. The reader can easily modify the arguments to handle the boundary in a different way.

The *reach* of a Voronoi region $V_T(x)$ is the distance between the site x and the furthest point inside $V_T(x)$, or, in other words, the radius of the smallest disc centered at x containing $V_T(x)$. We can estimate μ_{opt} as follows.

Lemma 3.1 *Let ℓ be the largest reach of any Voronoi region $V_T(t)$, for $t \in T$. Then*

$$\pi\ell^2/4 \leq \mu_{opt} \leq \pi\ell^2.$$

Proof. Let p be a point realizing the reach ℓ, that is, its distance to the nearest site is ℓ. It follows that $V_T(p)$ contains the disc with center p and radius $\ell/2$, and so $\mu(p) \geq \pi\ell^2/4$. The lower bound follows.

Let now x be the point realizing the optimal solution, that is $\mu(x) = \mu_{opt}$, and let $y \in V_T(x)$ be the point furthest from x. Its distance to the nearest site in T is at most ℓ, and so its distance to x is at most ℓ. It follows that $V_T(x)$ is contained in the disc with radius ℓ and center x, implying the upper bound. ◻

Note that the largest reach ℓ is also the radius of the largest empty circle. It can be computed in $O(n \log n)$ time by computing the Voronoi diagram of T and inspecting every vertex.

Our goal is to find a point x_{app} such that $\mu(x_{app}) \geq (1-\delta)\mu_{opt}$, for some parameter $\delta > 0$. We partition the unit square (containing T) into a grid of squares with side length ℓ. For each grid cell $\mathcal{Q}$, we will apply the ε-approximation technique of Section 2.1. We will define an estimate function e_S, such that for any $x \in \mathcal{Q}$ we have

$$|e_S(x) - \mu(x)| \leq \frac{\delta\pi\ell^2}{8},$$

and we pick a point $x_\mathcal{Q} \in \mathcal{Q}$ maximizing $e_S(x_\mathcal{Q})$. Let's first argue that this solves the problem: Let x_{app} be the point $x_\mathcal{Q}$ that maximizes $e_S(x_\mathcal{Q})$. Then

$$\mu(x_{app}) \geq e_S(x_{app}) - \frac{\delta\pi\ell^2}{8} \geq e_S(x_{opt}) - \frac{\delta\pi\ell^2}{8} \geq \mu_{opt} - \frac{\delta\pi\ell^2}{4} \geq \mu_{opt} - \delta\mu_{opt} = (1-\delta)\mu_{opt},$$

and so x_{app} is the desired approximate solution.

It remains to show how to define e_S and how to find the point $x_\mathcal{Q}$, for each grid cell $\mathcal{Q}$. Let's fix a grid cell $\mathcal{Q}$, and let x be a point in $\mathcal{Q}$. The reach of $V_T(x)$ is at most ℓ, and so $V_T(x)$ can intersect only $\mathcal{Q}$ itself and its eight neighboring grid cells. Consequently, all points of T participating in the definition of $V_T(x)$ lie in $\mathcal{Q}$ and the 24 grid cells at distance at most 2ℓ. Let $\mathcal{Q}'$ denote the union of these 25 grid cells, and let $T_\mathcal{Q} = T \cap \mathcal{Q}'$.

We make use of the following simple lemma.

Lemma 3.2 *Let S be a square grid of density ε in the plane, that is, the distance between neighboring grid points is ε, and let C be a convex body of diameter at most D. Then*

$$\left|\mu(C) - \varepsilon^2 \left|C \cap S\right|\right| \leq 4D\varepsilon^2.$$

Proof. Consider the tessellation of the plane into little squares of side length ε, where each point of S is the center of one little square. The boundary of C intersects at most $4D$ little squares, which implies the bound. ◻

We set $\varepsilon := \sqrt{\delta\pi\ell}/8$ and let S be a square grid of density ε, covering $\mathcal{Q}'$. For a point $x \in \mathcal{Q}$, let

$$e_S(x) = \varepsilon^2 \left|V_T(x) \cap S\right|$$

be the *estimate of the Voronoi region of* x. Making use of the fact that the diameter of $V_T(x)$ is at most 2ℓ, we then have by Lemma 3.2

$$|e_S(x) - \mu(x)| \leq 8\ell\varepsilon^2 \leq \delta\pi\ell^2/8,$$

and by what we observed above, it remains to find the point $x_{\mathcal{Q}} \in \mathcal{Q}$ maximizing $e_S(x_{\mathcal{Q}})$. To this end, we define

$$W(s) = \left\{ x \in \mathcal{Q} \;\middle|\; s \in V_T(x) \right\}.$$

Note that $W(s)$ is simply the largest disc with center s that contains no point of T in its interior, clipped to $\mathcal{Q}$. Let $\mathcal{W}_S = \{W(s) \mid s \in S\}$ and consider the arrangement $\mathcal{A}(\mathcal{W}_S)$. As in Section 2.1, our problem has reduced to finding a point in $\mathcal{Q}$ that is contained in the largest number of clipped discs in $\mathcal{W}_S$.

Theorem 3.3 *Given a set T of n points in the plane and a parameter $\delta > 0$, one can deterministically compute, in time $O(n/\delta^2 + n/\delta \log n)$, a point x_{app} such that $\mu(x_{app}) \geq (1-\delta)\mu_{opt}$.*

Proof. We start by computing the Voronoi diagram of T and inspecting its vertices to determine the largest reach ℓ. We then define the square grid, and determine the set of points $T_{\mathcal{Q}}$ relevant in each grid cell. Since a point of T is relevant in at most 25 grid cells, the total size of the sets $T_{\mathcal{Q}}$ is $O(n)$.

For each grid cell $\mathcal{Q}$ we take a square grid S of density $\varepsilon = \sqrt{\delta\pi\ell}/8$. It consists of

$$M = \frac{25\ell^2}{\varepsilon^2} = \frac{1600\ell}{\delta\pi}$$

points. For $s \in S$, the clipped disc $W(s)$ can be determined by finding the nearest neighbor to s in $T_{\mathcal{Q}}$. We do this by simply comparing the distance from s to each point in $T_{\mathcal{Q}}$. The arrangement $\mathcal{W}_S$ is computed by a sweep-line algorithm in time $O(M^2)$. The number of discs containing each face of the arrangement can again be determined by a simple transversal. We pick a point $x_{\mathcal{Q}}$ from this face, together with its estimate $e_S(x_{\mathcal{Q}})$.

By the choice of ℓ, every grid cell is within distance at most 2ℓ from a point of T. The number of grid cells handled is therefore at most $O(n)$. Each point of T appears at most $25M$ times in a nearest-neighbor computation, and so the overall running time is $O(n \log n + nM + nM^2) = O(n/\delta^2 + n \log n)$. ◻

4 Shape matching

Let P and Q be polygonal regions of complexity n and m, respectively, where $n \leq m$. For a vector x, let $V_{PQ}(x)$ denote $P(x) \cap Q$, where $P(x)$ is the region obtained by translating P by x, and let $\mu(x)$ denote the area of $V_{PQ}(x)$. As before, let μ_{opt} be $\max_x \mu(x)$, and let x_{opt} be such that $\mu_{opt} = x_{opt}$. We assume $\mu(Q) = 1$.

We proceed as in Section 2.2. We sample a set S of points in Q, and for a translation x (identified with a point x in the plane), we count the fraction that is in $P(x)$ to obtain the estimate

$$e_S = \frac{|V_{PQ}(x) \cap S|}{|S|}.$$

The following lemma is the equivalent of Lemma 2.3, with literally the same proof.

Lemma 4.1 *Let ν, δ be parameters, and let x be a translation such that $\mu(x) \geq \nu$ and $0 < \delta \leq 1$. Let S be a uniform sample from Q of size $M \geq c_1 \frac{\log n}{\delta^2 \nu}$, where c_1 is an appropriate constant. Then,*

$$\mathbf{Pr}[|e_S(x) - \mu(x)| > \delta \cdot \mu(x)] \leq \frac{1}{n^{10}}.$$

We now define, for $s \in S$, $W(s) = \{x \mid s \in P(x)\}$. Obviously, $W(s)$ is a copy of P, reflected at the origin and translated. Let $\mathcal{A}_P(S)$ be the arrangement of all regions $W(s)$. The following lemma is the equivalent of Lemma 2.4, and its proof is nearly identical.

Lemma 4.2 *Let ν, δ be parameters, let S be a uniform sample from Q of size $M \geq c_2 \frac{\log n}{\delta^2 \nu}$, where c_2 is an appropriate constant, and let x be a vertex of the arrangement $\mathcal{A}_P(S)$.*

1. *If $\mu(x) \leq \nu/4$, then $\mathbf{Pr}[|e_S(x)| \geq \nu/2] \leq \frac{1}{n^{10}}$.*

2. *If $\mu(x) \geq \nu/4$ then $\mathbf{Pr}[|e_S(x) - \mu(x)| > \delta \cdot \mu(x)] \leq \frac{1}{n^{10}}$.*

As in Section 2.2, we can now choose the vertex x_{app} of $\mathcal{A}_P(S)$ that maximizes $e_S(x_{app})$. Choosing $\nu = \delta = \varepsilon$, we find that $\mu(x_{app}) \geq \mu_{opt} - \varepsilon$ with probability at least $1 - 1/n^6$.

The complexity of the arrangement $\mathcal{A}_P(S)$ is $O(n^2 M^2) = O(n^2/\varepsilon^6 \log^2 n)$, and it can be computed within the same time bound. The vertex maximizing $e_S(x_{app})$ can be found by a simple traversal of this arrangement.

Consider now the case studied by de Berg et al. [dBGK+03]. Here, P is the union of n homothets of a convex body C of constant complexity, Q is the union of m homothets of C, the ratio between the largest and smallest homothet is bounded by a constant, and no point of the plane is contained in more than a constant number of homothets. The probabilistic analysis above goes through unchanged, so it remains to bound the complexity of the arrangement $\mathcal{A}_P(S)$. It consists of M translated copies of (the reflection of) P. The boundary of P has complexity n, as it is the union of n pseudodisks, and this boundary can be computed in time $O(n \log^2 n)$ [KLPS86]. Each boundary segment (belonging to a single homothet of C) can intersect only a constant number of other homothets in this set of nM homothets, and so the total number of vertices of $\mathcal{A}_P(S)$ is at most $O(nM^2)$. It can be computed in time $O(nM^2 \log n) = O(n/\varepsilon^6 \log^3 n)$, for instance by a plane sweep.

5 Conclusions

We have given a near-quadratic time algorithm for approximating the largest visible polygon inside a simple polygon. Our algorithm runs in near-linear time if the visibility polygon is reasonably large, a case that appears relevant in many applications. We also showed that approximating the area of the largest visible polygon is a 3SUM-hard problem [GO95], and as such it is unlikely to have a subquadratic algorithm.

In the second part of the paper, we applied a similar technique to the problem of finding the largest Voronoi cell one might occupy by a single point. Unlike the first problem, where direct random sampling does not yield any guaranteed approximation in the worst case, here direct random sampling seems to be possible. To do so, one has to prove that the area of the region

$$A = \left\{ x \;\middle|\; \mu\,(x) \geq (1-\delta)\mu_{opt} \right\}$$

is sufficiently large to be "hit" be a sample point. The bounds we were able to prove on the area of A result in an algorithm far slower than the one presented here, and used considerably more involved arguments. It seems that this problem is far from being well understood, and we leave it as open problem for further research. (See [DKS02] for related results.)

Acknowledgment

We thank Jeff Erickson for sketching out the proof of Lemma 2.10. We also thank Joseph S. B. Mitchell, Christian Knauer, René van Oostrum, and Helmut Alt for fruitful discussions regarding the problems addressed in this paper.

References

[ACC+01] H.-K. Ahn, S.-W. Cheng, O. Cheong, M. Golin, and R. van Oostrum. Competitive facility location along a highway. In *7th Annual International Computing and Combinatorics Conference*, volume 2108 of *LNCS*, pages 237–246, 2001.

[AFRW98] H. Alt, U. Fuchs, G. Rote, and G. Weber. Matching convex shapes with respect to the symmetric difference. *Algorithmica*, 21:89–103, 1998.

[AS00] N. Alon and J. H. Spencer. *The probabilistic method.* Wiley Inter-Science, 2nd edition, 2000.

[CHLM02] O. Cheong, S. Har-Peled, N. Linial, and J. Matoušek. The one-round Voronoi game. In *Proc. 18th Annu. ACM Sympos. Comput. Geom.*, pages 97–101, 2002.

[CS89] K. L. Clarkson and P. W. Shor. Applications of random sampling in computational geometry, II. *Discrete Comput. Geom.*, 4:387–421, 1989.

[dBCD+98] M. de Berg, O. Cheong, O. Devillers, M. van Kreveld, and M. Teillaud. Computing the maximum overlap of two convex polygons under translations. *Theo. Comp. Sci.*, 31:613–628, 1998.

[dBGK+03] M. de Berg, P. Giannopoulos, C. Knauer, R. van Oostrum, and R. C. Veltkamp. Maximizing the area of overlap of two unions of convex homothets under translation. Manuscript, 2003.

[DKS02] F. Dehne, R. Klein, and R. Seidel. Maximizing a Voronoi region: The convex case. In *Proc. 13th Annu. Internat. Sympos. Algorithms Comput.*, pages 624–634, 2002.

[EH02] A. Efrat and S. Har-Peled. Locating guards in art galleries. *2nd IFIP Internat. Conf. Theo. Comp. Sci.*, pages 181–192, 2002.

[Gho87] S. K. Ghosh. Approximation algorithms for art gallery problems. In *Proc. Canadian Inform. Process. Soc. Congress*, 1987.

[GMMN90] L. Gewali, A. Meng, J. S. B. Mitchell, and S. Ntafos. Path planning in $0/1/\infty$ weighted regions with applications. *ORSA J. Computing*, 2:253–272, 1990.

[GO95] A. Gajentaan and M. H. Overmars. On a class of $O(n^2)$ problems in computational geometry. *Comput. Geom. Theory Appl.*, 5:165–185, 1995.

[HP98] D. Hochbaum and A. Pathria. Analysis of the greedy approach in covering problems. *Naval Research Quarterly*, 45:615–627, 1998.

[KLPS86] K. Kedem, R. Livne, J. Pach, and M. Sharir. On the union of Jordan regions and collision-free translational motion amidst polygonal obstacles. *Discrete Comput. Geom.*, 1:59–71, 1986.

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, New York, NY, 1995.

[NT94] S. C. Ntafos and M. Z. Tsoukalas. Optimum placement of guards. *Info. Sciences*, 76:141–150, 1994.

[OBSC00] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu. *Spatial tessellations: Concepts and applications of Voronoi diagrams.* Probability and Statistics. Wiley, 2nd edition edition, 2000.

[OPP03] The open problems project. Problem 33: Sum of square roots. http://cs.smith.edu/~orourke/TOPP/P33.html, 2003.

[O'R87] J. O'Rourke. *Art Gallery Theorems and Algorithms.* The International Series of Monographs on Computer Science. Oxford University Press, New York, NY, 1987.

[Urr00] J. Urrutia. Art gallery and illumination problems. In J. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 973–1027. North-Holland, 2000.

[Val98] P. Valtr. Guarding galleries where no point sees a small area. *Israel J. Math*, 104:1–16, 1998.