Cutting Polygons into Small Pieces with Chords: Laser-Based Localization

Esther M. Arkin
Stony Brook University, NY, USA
esther.arkin@stonybrook.edu

Rathish Das
Stony Brook University, NY, USA
rathish.das@stonybrook.edu

Jie Gao
Rutgers University, Piscataway, NJ, USA
jg1555@rutgers.edu

Mayank Goswami
Queens College of CUNY, New York, NY, USA
mayank.goswami@qc.cuny.edu

Joseph S. B. Mitchell
Stony Brook University, NY, USA
joseph.mitchell@stonybrook.edu

Valentin Polishchuk
Linköping University, Norrköping, Sweden
valentin.polishchuk@liu.se

Csaba D. Tóth
California State University Northridge,
Los Angeles, CA, USA
Tufts University, Medford, MA, USA
csaba.toth@csun.edu

Abstract

Motivated by indoor localization by tripwire lasers, we study the problem of cutting a polygon into small-size pieces, using the chords of the polygon. Several versions are considered, depending on the definition of the "size" of a piece. In particular, we consider the area, the diameter, and the radius of the largest inscribed circle as a measure of the size of a piece. We also consider different objectives, either minimizing the maximum size of a piece for a given number of chords, or minimizing the number of chords that achieve a given size threshold for the pieces. We give hardness results for polygons with holes and approximation algorithms for multiple variants of the problem.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis; Theory of computation → Packing and covering problems; Mathematics of computing → Combinatorial algorithms; Theory of computation → Computational geometry

Keywords and phrases Polygon partition, Arrangements, Visibility, Localization

Digital Object Identifier 10.4230/LIPIcs.ESA.2020.7

Related Version A full version of the paper is available at `https://arxiv.org/abs/2006.15089`.

Funding This research was partially supported by NSF grants CCF-1439084, CCF-1540890, CCF-1617618, CCF-1716252, CCF-1725543, CCF-1910873, CCF-2007275, CNS-1618391, CNS-1938709, CRII-1755791, CSR-1763680, DMS-1737812, DMS-1800734, and OAC-1939459. The authors also acknowledge partial support from the US-Israel Binational Science Foundation (project 2016116), the DARPA Lagrange program, the Sandia National Labs and grants by the Swedish Transport Administration and the Swedish Research Council.

Acknowledgements We thank Peter Brass for technical discussions and for organizing an NSF-funded workshop where these problems were discussed and this collaboration began.

1 Introduction

Indoor localization is a challenging and important problem. While GPS technology is very effective outdoors, it generally performs poorly inside buildings, since GPS depends on line-of-sight to satellites. Thus, other techniques are being considered for indoor settings. One of the options being investigated for localization and tracking is to use one-dimensional

28th Annual European Symposium on Algorithms (ESA 2020).
Editors: Fabrizio Grandoni, Grzegorz Herman, and Peter Sanders; Article No. 7; pp. 7:1–7:23
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

tripwire sensors [16] such as laser beams, video cameras with a narrow field of view [31], and pyroelectric or infrared sensors [12, 14]. In these approaches, multiple sensors emitting directional signal beams are deployed in an environment, with the beams inducing an arrangement that cuts the domain into cells, allowing one to track the movement of a mobile target from one cell to another when it crosses the signal beam. Since the accuracy of the localization depends on the sizes of the cells, it is desirable to cut the polygon into *small* pieces. With such beam deployment, one can also ensure that no "large" object can be "hidden" in the domain, since any such object will necessarily intersect one of the beams.

In the literature there have been studies of target localization and tracking using such "tripwire" sensors. Zheng, Brady, and Agarwal [32] consider general models of "boundary sensors" that are triggered when an object crosses them. They assume that the position of the sensors is already given and consider the signal processing problem of determining the location and trace of a target by the spatial and temporal sequence of the laser beams crossed by the target. In this paper, we focus on the problem of optimizing the placement of signal beam sensors to minimize the ambiguity of target location within each cell.

Problem Formulation and Notation. We study various versions of the laser cutting problem. The input polygon, denoted by P, is a closed polygonal domain (i.e., a connected compact set in $\mathbb{R}^2$ with piecewise linear boundary) having a total of n vertices, r of which are reflex (having internal angle greater than π). The terms "cut" and "laser" will be used interchangeably to denote a chord of P, i.e., a maximal line segment in P whose relative interior lies in the interior of P. The *measure* (or *size*) of a cell in the arrangement will be (a) the cell's area, (b) its diameter (defined as the maximum Euclidean distance between two points of the cell), or (c) the radius of the largest inscribed disk within the cell.

For each measure, we consider two formulations of the optimization problem:

- MinMeasure: Given a positive integer k, determine how to place k laser beams in P to minimize the maximum measure, δ, of a cell in the arrangement of the lasers.
- Min-LaserMeasure: Given $\delta > 0$, determine the smallest number of laser beams to cut P into cells each of measure at most δ.

In Min-LaserMeasure, no generality is lost by taking the cell size bound, δ, to be 1. We assume that the optimal solution is greater than a constant c; otherwise, the problem can be solved optimally in $O(n^{\text{poly}(c)})$ time (in the real RAM model of computation, standard for geometric algorithms) by reducing it to a mathematical program whose variables are the locations of the lasers endpoints on the boundary of P (the space of the variables would be split into regions of fixed combinatorial types for all the lasers, and in each region, the measures for the cells of the partition of P will be explicitly written and optimized – since each cell has poly$(c) = O(1)$ complexity, the optimization problem will be of constant size). It may be interesting to investigate also the opposite scenario and obtain efficient algorithms for minimizing the measures using a small given number of lasers. Further variants of the problem may be defined. One possible requirement is to use only axis-aligned lasers – in fact, with this restriction (of primarily theoretical interest) we obtain better approximations than for the more general case of unrestricted-orientation lasers.

Results. We give hardness results and approximation algorithms for several variants of the problems, using a variety of techniques. Specifically,

- Section 2 proves hardness of our problems in polygons *with holes*: we show that it is NP-hard to decide whether one can split the domain into pieces of measure at most δ, using a given number k of lasers (this holds for any of the measures, which implies that both MinMeasure and Min-LaserMeasure are hard for polygons with holes). Our hardness reductions hold using axis-parallel lasers, as well, which implies that the problem is hard with or without the restriction to axis-aligned lasers.

- Section 3.1 gives an $O(\log r)$-approximation for Min-LaserArea in *simple* polygons. The algorithm "unrefines" the ray shooting subdivision by Hershberger and Suri [17], merging the triangles bottom-up along the decomposition tree; the merging stops whenever the next merge would create a cell of area greater than δ, implying that the boundaries between the merged cells can be charged to disjoint parts of P of area more than δ. The lasers are then put along the cell boundaries of the coarsened subdivision; since the subdivision is obtained by cutting out $O(1)$ children from parents in a tree on the original subdivision (where the children were separated from parents by polygonal chains of $O(1)$ complexity), we can charge these $O(1)$ lasers to the intersection of OPT with a region of area more than δ. The remaining large pieces in the coarsened subdivision (e.g., triangles of area more than δ in the initial triangulation) are cut with a suitable grid of lasers, which is within a constant factor of optimal subdivision for each piece. The $O(\log n)$ approximation factor then follows from the fact that each laser could pass through $O(\log n)$ cells of the original subdivision (the subdivision's core property). To bring the approximation factor down to $O(\log r)$ we decompose P into convex pieces with a decomposition whose stabbing number is $O(\log r)$ (a result, which may be of independent interest) and use the same scheme as with the Hershberger–Suri decomposition.
- In Section 3.2 we present a bi-criteria approximation to the diameter version for *simple* polygons: if k lasers can cut P into pieces of diameter at most δ, we find a cutting with at most $2k$ lasers into $O(\delta)$-diameter pieces. In Section 3.3 we use the bi-criteria algorithm to give a constant-factor approximation to MinDiameter. Both algorithms use only axis-aligned lasers, yielding the same approximation guarantees for the versions with general-direction lasers and with axis-aligned lasers.
- Section 4 gives a constant-factor approximation to Min-LaserDiameter and Min-LaserArea in *simple* polygons under the restriction that the lasers are axis-aligned. The algorithms are based on "histogram decomposition" with constant stabbing number and solving the problems in each histogram separately.
- In Section 5 we give a bi-criteria approximation to the diameter version in polygons *with holes* under the restriction that lasers are axis-parallel. The algorithm is similar to the one for simple polygons in that they both use a grid; however, everything else is different: in simple polygons we place lasers along grid lines, while in polygons with holes the grid lines just subdivide the problem (in fact, we consider the vertical and the horizontal strips separately). More importantly, even though we place axis-aligned lasers in both simple and nonsimple polygons, for the former we approximate cutting with arbitrary-direction lasers, while for the latter only cuttings with axis-aligned lasers (approximating cuttings with general-direction lasers in polygons with holes is open). We use the bi-criteria algorithm to give a constant-factor approximation to MinDiameter in polygons with holes – this part is the same as for simple polygons.
- Section 6 gives an $O(\log \mathrm{OPT})$-approximation for Min-LaserCircle in polygons *with holes*. The algorithm is based on a reduction to the SetCover problem.

Table 1 summarizes our results. The running times of our algorithms depend on the output complexity, which may depend on the size (area, perimeter, etc.) of P. Some of our algorithms can be straightforwardly made to run in strongly-polynomial time, producing a strongly-polynomial-size representation of the output; for others, such conversion – which in general is outside our scope – is not easily seen. Many versions of the problem still remain open. For simple polygons, despite considerable attempts, we have neither hardness results nor polynomial-time algorithms to compute an optimal solution; all of our positive results are approximation algorithms.

Table 1 Approximations for simple polygons. The results marked with asterisks apply also to polygons with holes (either directly or with a similar/extended algorithm).

	Axis-Parallel Lasers		Unrestricted-Direction Lasers	
	Min-LaserMeasure	MinMeasure	Min-LaserMeasure	MinMeasure
Area	$O(1)$ § 4	OPEN	$O(\log r)$ § 3.1	OPEN
Diameter	$O(1)$ § 4	$O(1)$* § 3.3, § 5	bi-critreria § 3.2	$O(1)$ § 3.3
In-circle radius	$O(\log \mathrm{OPT})$* § 6	OPEN	$O(\log \mathrm{OPT})$* § 6	OPEN

Related Previous Work. Previous results on polygon decomposition [21] use models that do not support laser cuts or are restricted to convex bodies. For example, *Borsuk's conjecture* [5, 18, 19] seeks to partition a convex body of unit diameter in $\mathbb{R}^d$ into the minimum number of pieces of diameter less than one. Conway's *fried potato problem* [3, 9] seeks to minimize the maximum in-radius of a piece after a given number of *successive* cuts by hyperplanes for a convex input polyhedron in $\mathbb{R}^d$. Croft et al. [9, Problem C1] raised a variant of the problem in which a convex body is partitioned by an *arrangement* of hyperplanes (i.e., our problem in $\mathbb{R}^d$), but no results have been presented.

Equipartition problems ask to partition convex polygons into convex pieces all having the same area or the same perimeter (or other measures) [2, 4, 20, 22, 27, 29]. In these problems, the partition is not restricted to chords (or hyperplanes). Topological methods are used for existential results in this area, and very few algorithmic results are known [1]. Another related problem is the family of so-called *cake cutting problems* [13, 28], in which an infinite straight line "knife" is used to cut a convex "cake" into (convex) pieces that represent a "fair" division into portions. In contrast, we are interested in cutting *nonconvex* polygons into connected pieces.

In [6] several variants of Chazelle's result from [8] were explored, including cutting the polygon along a chord to get approximately equal areas of the two resulting parts. Yet another related problem is that of "shattering" with arrangements [11], in which one seeks to isolate objects in cells of an arrangement of a small number of lines, but without consideration of the size of the cells (as is important in our problem).

2 Hardness in Polygons with Holes

We show that for all three measures (area, diameter, the radius of the largest inscribed circle) it is NP-hard to decide whether a given polygon P *with holes* can be divided into pieces of small measure using a given number of lasers, both for unrestricted-orientation and axis-aligned lasers. However, it is currently open whether these problems remain NP-hard for simple polygons.

We prove hardness by reduction from the 3SAT problem. Our polynomial-time reduction is similar to previous reductions for line cover problems, which are geometric variants of set cover [23]. In particular, Megiddo and Tamir [25] proved that the LineCover problem is NP-complete: Given n points in the plane and an integer k, decide whether the points can be covered by k lines. Hassin and Megiddo [15] proved hardness for MinimumHittingHorizontalUnitSegments problem: Given n horizontal line segments in the plane, each of unit length, and an integer k, decide whether there exists a set of k axis-parallel lines that intersects all n segments. Our reduction is based on the idea of Hassin and Megiddo, but requires some adjustments to generate a subdivision of a polygon.

▶ **Theorem 1.** *In a polygon with holes, both* MinArea *and* Min-LaserArea *are NP-hard (with or without the axis-aligned lasers restriction).*

Proof. We reduce from 3-SAT. Let Φ be a boolean formula in 3CNF with m clauses $c_1, \ldots, c_m$, and n variables $x_1, \ldots, x_n$. We construct an orthogonal polygon P with holes and an integer k such that Φ is satisfiable if and only if P can be subdivided into regions of area at most 2 using k lasers. (The reduction goes through with or without the restriction that all lasers are axis-parallel).

We construct a polygon P from the rectangle $B = [0, 7m + 2] \times [0, 3n + 4]$ by carving rectangular "rooms" connected by narrow corridors. The rooms are pairwise disjoint and they each have area of 2. The corridors are axis-parallel, run between opposite sides of the bounding box B, and their width is $1/(100 \max\{m, n\})$. See Fig. 1 for an illustration.

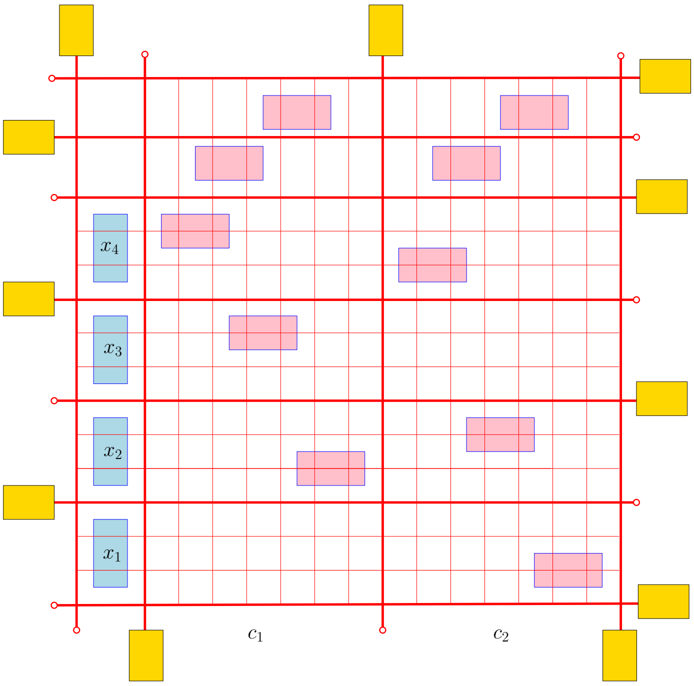

Figure 1 An example for the rooms and corridors for $\Phi = (x_2 \vee \overline{x}_3 \vee \overline{x}_4) \wedge (x_1 \vee \overline{x}_2 \vee x_4)$. The rooms corresponding to variables are in blue. The rooms corresponding to clauses (five rooms per clause) are in pink. The corridors are shown in red, some of which are connected to additional rooms (shown in yellow). The polygon is composed of the union of all the rooms and corridors.

Variable rooms. For each variable x_i, $i = 1, \ldots, n$, create one room: $[\frac{1}{2}, \frac{3}{2}] \times [3(i-1) + \frac{1}{2}, 3i - \frac{1}{2}]$. Note that all rooms are to the left of the line $x = 2$.

Clause rooms. For each clause c_j, $j = 1, \ldots, m$, create five rooms. All five rooms have size $2{\times}1$ and lie between the lines $x = 7(j-1)+2$ and $x = 7j+2$. Three out of five rooms are aligned with the variable rooms. Suppose c_j contains the variables x_i, $x_{i'}$, and $x_{i''}$, where $i < i' < i''$. If x_i is nonnegated, then create the room $[7(j-1) + \frac{1}{2}, 7(j-1) + \frac{5}{2}] \times [3(i-1) + \frac{1}{2}, 3i - \frac{3}{2}]$;

otherwise create the room $[7(j-1)+\frac{1}{2}, 7(j-1)+\frac{5}{2}]\times[3(i-1)+\frac{3}{2}, 3i-\frac{1}{2}]$. We create a room for $x_{i'}$ (resp., $x_{i''}$) analogously, shifted by a horizontal vector $(0,2)$ (resp;., $(0,4)$). Note that the x-projections of these rectangles do not overlap. Two additional rooms lie above the variable rooms: $[7(j-1)+\frac{3}{2}, 7(j-1)+\frac{7}{2}]\times[2n+\frac{1}{2}, 2n+\frac{3}{2}]$ and $[7(j-1)+\frac{7}{2}, 7(j-1)+\frac{11}{2}]\times[2n+\frac{5}{2}, 2n+\frac{7}{2}]$.

Corridors and separator gadgets. Create narrow corridors along the vertical lines $x = 0, 2, 3, \ldots, 7m$ and horizontal lines $y = 0, 1, 2, \ldots, 3n$, $y = 3n + 2$, and $y = 3n + 4$. Add rectangular rooms of area 2 at one end of some of the corridors. Specifically, we add rooms to the corridors at $x = 0$ and $x = 7j + 2$ for $j = 0, 1, \ldots, m$ alternately at the top and bottom endpoints; and similarly for the corridors at $y = 3i$ for $i = 0, 1, \ldots, n$, $y = 3n + 2$, and $y = 3n + 4$, alternately at the left and right endpoints. Altogether, $m + n + 5$ corridors have rooms at their endpoints.

Finally, we set the parameter $k = 3m + 2n + 5$. This completes the description of an instance corresponding to the Boolean formula Φ.

Equivalence. Let $\tau : x_i \to \{\texttt{true}, \texttt{false}\}$ be a satisfying truth assignment for Φ. We show that P can be subdivided by k lasers into regions of area at most 2. Place lasers at all horizontal and vertical lines that have additional rooms at their endpoints; this requires $m + n + 5$ lasers. These lasers subdivide P into subpolygons that each intersect at most one room. For $i = 1, \ldots, n$, if $\tau(x_i) = \texttt{true}$, then place a horizontal laser at $y = 3(i - 1) + 1$ (along the bottom corridor touching room for x_i), otherwise at $y = 3(i - 1) + 2$ (along the top corridor touching room for x_i). These lasers split each variable room into two rectangles of area $\frac{1}{2}$ and $\frac{3}{2}$. For $j = 1, \ldots, m$, we place two vertical lasers that subdivide the rooms associated with clause c_j. Since τ is a satisfying truth assignment, the rooms corresponding to true literals are already split by horizontal lasers. As can easily be checked, the remaining (at most 4) rooms can be split using two vertical lasers. Now P is subdivided into pieces that each intersect at most one room, and contains at most 1.5 area of each room. Since the corridors are narrow, the area of each piece is less than 2, as required. We have used n horizontal lasers for the variables, and $2m$ vertical lasers for clauses. Overall, we have used $(m + n + 5) + n + 2m = 3m + 2n + 5$ lasers.

Suppose now that $k = 3m + 2n + 5$ lasers can subdivide P into polygons of area at most 2. We show that Φ is satisfiable. The area of each room is about 2, so they each intersect at least one laser. Each variable room requires at least one laser; and the n variable rooms jointly require n lasers (as no laser can intersects two variable rooms). Each clause is associated with two rooms above the line $y = 3n$; which jointly require two lasers. Overall these rooms require $2m$ lasers.

Note that a laser that intersects a clause rooms above $y = 3n$ or a variable room cannot intersect any room at the end of corridors. We are left with at most $k - (n + 2m) = m + n + 5$ lasers to split these rooms. Since we have precisely $m + n + 5$ rooms at the end of the corridors, and no laser can intersect two such rooms, there is a unique laser intersecting each of these rooms. As argued above, for $i = 1, \ldots, n$, the room associated with x_i intersects only one laser. If this laser intersects the corridor at $y = 3(i - 1) + 1$, then let $\tau(x_i) = \texttt{true}$, otherwise $\tau(x_i) = \texttt{false}$. For $j = 1, \ldots, m$, there are two lasers that intersect the two rooms associated with c_j above $y = 3n$. These two lasers cannot intersect all three rooms associated with c_j below $y = 3n$. Consequently, at least one of these rooms intersects a laser coming from a variable room. Hence each clause contains a true literal, and Φ is satisfiable. ◀

The proofs of the following two theorems are presented in the full version of the paper.

▶ **Theorem 2.** *In polygons with holes, both* MinDiameter *and* Min-LaserDiameter *are NP-hard (with or without the axis-aligned lasers restriction).*

▶ **Theorem 3.** *In polygons with holes, both* MinCircle *and* Min-LaserCircle *are NP-hard (with or without the axis-aligned lasers restriction).*

3 Decomposition Algorithms for Simple Polygons

In this section, we present approximation results for decomposing a simple polygon P by lasers of arbitrary orientations (recall that n denotes the total number of vertices of P and r is the number of reflex vertices). We describe an $O(\log r)$-approximation for Min-LaserArea (Section 3.1), a bi-criteria algorithm for diameter (Section 3.2), and a $O(1)$-approximation for MinDiameter (Section 3.3).

3.1 Min-LaserArea

Given a simple polygon P and a threshold δ, we wish to find the minimum number of lasers that subdivide P into pieces, each of area at most 1. We start with the easy $O(1)$-approximation in the special case when P is a convex polygon (Proof in the full version).

▶ **Lemma 4.** *For every convex polygon P, we can find a set of $k = O(\sqrt{\text{area}(P)})$ lasers that subdivide P into pieces, each of area at most 1, in $O(k+n)$ time. Every decomposition into pieces of area at most 1 requires $\Omega(\sqrt{\text{area}(P)})$ lasers.*

Overview. We give a brief overview of our approximation algorithm for a simple polygon P. The basic idea is to decompose P into convex pieces, and use Lemma 4 to further decompose each convex piece. There are two problems with this naïve approach: (1) a laser in an optimal solution may intersect several convex pieces (i.e., the sum of lower bounds for the convex pieces is not a global lower bound); and (2) the lasers used for a convex decomposition are not accounted for. We modify the basic approach to address both of these problems.

We use the Hershberger–Suri triangulation (as a convex subdivision). For a simple polygon P with n vertices, Hershberger and Suri [17] construct a Steiner triangulation into $O(n)$ triangles such that every chord of P intersects $O(\log n)$ triangles. We can modify their construction to produce a Steiner decomposition into a set $\mathcal{C}$ of convex cells (rather than triangles) such that each laser intersects $O(\log r)$ convex cells, where r is the number of reflex vertices of P. Thus, each laser of OPT can help partition $O(\log r)$ convex cells; this factor dominates the approximation ratio of our algorithm.

A convex cell $C \in \mathcal{C}$ is *large* if $\text{area}(C) > 1$, otherwise it is *small.* We decompose each large convex cell using Lemma 4. We can afford to place $O(1)$ lasers along the boundary of a large cell. We cannot afford to place lasers on the boundaries of all small cells. If we do not separate the small cells, however, they could merge into a large (nonconvex) region, so we need *some* separation between them. In the algorithm below, we construct such separators recursively by carefully unrefining the Hershberger–Suri triangulation. The unrefined subdivision is no longer a triangulation, but we maintain the properties that (i) each cell is bounded by $O(1)$ lasers within each pseudotriangle (and an arbitrary number of consecutive edges of P), and (ii) every chord of P intersects $O(\log n)$ cells.

Basic properties of the Hershberger–Suri triangulation. Given a simple polygon P with n vertices, Hershberger and Suri [17] construct a Steiner-triangulation in two phases (see Fig. 2 for an example): First, they subdivide P into $O(n)$ pseudotriangles (i.e., simple

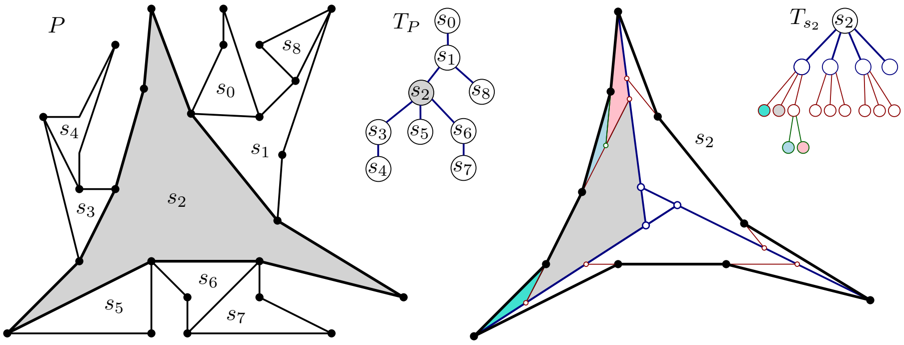

Figure 2 Left: A simple polygon P, decomposed into pseudotriangles, and the dual graph T_P. Right: A pseudotriangle s_2 is recursively subdivided into Steiner triangles, with recursion tree T_{s_2}.

polygons with precisely three convex vertices) using $O(n)$ noncrossing diagonals of P; and then subdivide each pseudotriangle into Steiner triangles. The runtime of their algorithm, as well as the number of Steiner triangles, is $O(n)$. Let $\mathcal{S}$ denote the set of pseudotriangles produced in the first phase; and let T_P be the *dual tree* of the pseudotriangles, in which each node corresponds to a pseudotriangle, and two nodes are adjacent if and only if the corresponding pseudotriangles share an edge (a diagonal of P). Note that the degree of T_P is not bounded by a constant (it is bounded by n), as a pseudotriangle may be adjacent to arbitrarily many other pseudotriangles. We consider T_P to be a rooted tree, rooted at an arbitrary pseudotriangle. Then every nonroot pseudotriangle s in $\mathcal{S}$ has a unique edge incident to the parent of s; we call this edge the *parent edge of s*.

Hershberger and Suri subdivide each pseudotriangle $s \in S$ recursively: In each step, they use $O(1)$ line segments to subdivide a pseudotriangle into $O(1)$ pseudotriangles, which are further subdivided recursively until they obtain triangles. Let us denote by T_s the recursion tree for s. Each vertex $v \in T_s$ represents a region $R_v \subset s$: The root of T_s represents s, and the leaves represent the Steiner triangles in s. The recursive subdivision maintains the following two properties: (a) Every edge of s is incident to a unique region in each level of T_s, (b) For each node $v \in T_s$, the boundary between R_v and $s \setminus R_v$ is a polyline with $O(1)$ edges (that is, R_v is bounded by $O(1)$ line segments inside s, and some sequence of consecutive edges of s).

Algorithm. We are ready to present an approximation algorithm for Min-LaserArea. Given a simple polygon Q, we begin by computing the Hershberger–Suri triangulation, the pseudotriangles $\mathcal{S}$, the dual tree T_P, and a recursion tree T_s for each pseudotriangle $s \in \mathcal{S}$. We then process the pseudotriangles in a bottom-up traversal of T_P.

Within each pseudotriangle $s \in \mathcal{S}$, we unrefine the Steiner triangulation of s by merging some of the cells into one cell (the resulting larger cells need not be triangular or convex). Initially, each node $v \in T_s$ corresponds to a region $R_v \subseteq s$. However, if we do not place lasers along the edges of s, then R_v may be adjacent to (and merged with) other cells that are outside the pseudotriangle s, along the boundary of s. Since we have an upper bound on the total area of each cell in the final decomposition, we need to keep track of the area of the region on both sides of an edge of the pseudo-triangulation. In the course of unrefinement algorithm for all $s \in \mathcal{S}$, we compute nonnegative *weights* $w(\cdot)$ for all edges of the pseudotriangulation. The weights are used for bookkeeping purposes. Specifically, the edges of P have zero weight. In a bottom-up traversal of T_P, when we start processing a pseudotriangle s, the *weights* $w(e)$

have already been computed for all edges of the pseudo-triangle s except the parent edge of s. The weight $w(e)$ for the parent edge e of s is determined when we have computed the unrefined subdivision of s; and $w(e)$ will be the area of the unrefined cell in s adjacent to the parent edge. A node $v \in T_s$ initially corresponds to a region R_v within the pseudotriangle s, but in the final decomposition of P, the node is part of some larger cell $\widehat{R}_v \subseteq P$, with $\text{area}(\widehat{R}_v) = \text{area}(R_v) + \sum_e w(e)$, where the summation is over all edges of s on the boundary of R_v, and $w(e)$ denotes the area of the cell on the opposite side of e.

As the weight of the parent edge is not available yet when we unrefine s, we modify the recursion tree T_s as follows: We choose the root to be the leaf $v_0 \in T_s$ adjacent to the parent edge of s, and reverse the parent-child relation on all edges of T_s along the s-v_0 path. We denote the modified recursion tree T'_s (Fig. 3 (left)). For all nodes v along the s-v_0 path (including s and v_0), we redefine the corresponding regions of the nodes in T'_s as follows. We denote by $R_v(T_s)$ and $R_v(T'_s)$ the regions corresponding to node v in trees T_s and T'_s, respectively. We set $R_{v_0}(T'_s) := s$ and for all other nodes v along the s-v_0 path (including s), we set $R_v(T'_s) := s \setminus R_u(T_s)$, where u is the parent of v in T'_s. With a slight abuse of notation, we set $R_v = R_v(T'_s)$ for all $v \in T'_s$ for the remainder of the algorithm. Note that $\text{area}(R_v)$ monotonically decreases with the depth in T'_S.

In a bottom-up traversal of T_P, consider every $s \in \mathcal{S}$. We proceed with two phases (see Fig. 3 for an example).

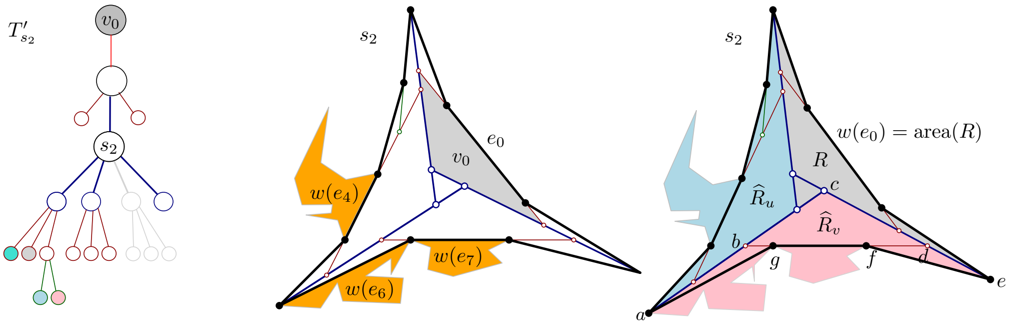

Figure 3 Left: The modified recursion tree T'_{s_2}. Middle: pseudotriangle s_2 with the initial Steiner triangulation, edge weights representing the areas of adjacent regions in the descendants of s_2, and the parent edge e_0 of s_2. Right: The unrefined subdivision of s_2 into R_u, R_v, and R; larger cells $\widehat{R}_u$ and $\widehat{R}_v$ (blue and pink), and the weight $w(e_0) = \text{area}(R)$ of the parent edge of s_2 (gray).

Phase 1 of the algorithm is an unrefinement process, that successively merges small cells of the Hershberger–Suri triangulation (no lasers are involved). We initialize three variables:

$$R := s, \qquad T := T'_s, \qquad U_s := \emptyset,$$

where $R \subseteq s$ is the region yet to be handled, T is a subtree of T'_s corresponding to the region R, and U_s is the set of interior-disjoint faces in s produced by the unrefinement process. While $\text{area}(R) > 1$, do the following:

- Find a lowest node $v \in T$ for which $\text{area}(\widehat{R}_v) > 1$,
- Set $U_s := U_s \cup \{\widehat{R}_v\}$,
- Set $R := R \setminus R_v$,
- Delete the subtree rooted at v from T, and
- For all ancestors u of v, set $\widehat{R}_u := \widehat{R}_u \setminus \widehat{R}_v$.

When the while loop ends, define the weight of the parent edge of s to be $\text{area}(R)$.

Phase 2 of the algorithm positions lasers in a pseudotriangle s as follows.

For every region $\widehat{R}_v \in U_s$, do:

- Step 1. Place lasers along all edges of the boundary between $\widehat{R}_v$ and $s \setminus \widehat{R}_v$, and the boundaries between R_v and $R_{v'}$ for all children v' of v. For example in Fig. 3 (right), two lasers are placed along the edges (a, c) and (c, e) that disconnect $\widehat{R}_v$ from s_2. Also, a laser that is placed along edge (b, d) that separates the children of R_v.
- Step 2. If $\text{area}(R_v) \geq 1$ (which means R_v has not merged with any other region in Phase 1, i.e., $\widehat{R}_v = R_v$ hence $\widehat{R}_v$ is convex), subdivide $\widehat{R}_v$ by $\Theta(\sqrt{\text{area}(R_v)})$ lasers according to Lemma 4.

This completes the description of our algorithm.

▶ **Theorem 5.** *Let P be a simple polygon with n vertices, and let k^* be the minimum number of lasers that subdivide P into pieces of area at most 1. We can find an integer k with $k^* \leq k \leq O(k^* \log n)$ in $O(n)$ time, and a set of k lasers that subdivide P into pieces of area at most 1 in output-sensitive $O(k + n)$ time.*

Proof. Phase 1 of our algorithm (unrefinement) subdivides each pseudotriangle $s \in \mathcal{S}$ into regions such that each region corresponds to a subtree rooted at some node v of the recursion tree T'_s. Node v corresponds to a region $R_v \subset s$, and a possibly larger region $\widehat{R}_v \subset P$ which is the union of R_v and adjacent regions in the descendant pseudotriangles of s adjacent to R_v. Phase 1 of the algorithm ensures that $\text{area}(\widehat{R}_v) > 1$ (therefore, $\widehat{R}_v$ must intersect at least one laser in OPT), but for all children v' of v in T'_s, we have $\text{area}(\widehat{R}_{v'}) \leq 1$.

In Step 1, the algorithm uses $O(1)$ lasers for each $v \in U_s$ to separate $\widehat{R}_v$ from $s \setminus \widehat{R}_v$. Recall that the recursion tree T_s has bounded degree. Consequently, we use $O(1)$ lasers to separate $\widehat{R}_{v'}$ from $\widehat{R}_v \setminus \widehat{R}_{v'}$ for all children v' of v. These polylines subdivide $\widehat{R}_{v'}$ into smaller regions of area at most 1. Overall, we have used $O(1)$ lasers for each of these nodes $v \in T'_s$, $s \in \mathcal{S}$. Note that each region $\widehat{R}_v$ is the union of triangles from the Hershberger–Suri Steiner triangulation, and so each laser in OPT intersects $O(\log n)$ such regions. Consequently, we use $O(k^* \log n)$ lasers in Step 1.

Finally, consider the lasers used in Step 2 for subdividing the triangles $t \in T$ with $\text{area}(t) > 1$. By Lemma 4, each such triangle intersects at least $\Omega(\sqrt{\text{area}(t)})$ lasers in any valid solution; and conversely each laser of an optimal solution intersects $O(\log n)$ regions in T. Consequently, the number of lasers uses in Step 2 is $\sum_{t \in T} O(\sqrt{\text{area}(t)}) \leq O(k^* \log n)$.

It remains to show that the algorithm runs in $O(n + k)$ time. The Hershberger-Suri Steiner triangulation can be computed in $O(n)$ time [17]. It consists of $O(n)$ triangles, hence the combined size of the dual tree T_p, and the recursion trees T_s, $s \in \mathcal{S}$, is also $O(n)$. The unrefinement algorithm is done in a single traversal of these trees, spending $O(1)$ time at each node. For each large cell (triangle) of the Hershberger-Suri triangulation, by Lemma 4, we can compute a minimum bounding box and the *number* of lasers used by the algorithm in $O(1)$ time. Computing all k lasers requires $O(k)$ additional time. ◀

An $O(\log r)$ Approximation for Min-LaserArea in Simple Polygons. We can improve the approximation ratio in Theorem 5 from $O(\log n)$ to $O(\log r)$, where r is the number of reflex vertices of P, if we replace the Hershberger–Suri triangulation with a convex decomposition. (Hershberger and Suri decompose P into *triangles* to support ray shooting queries, but for our purposes a decomposition into convex cells suffices.)

Let $(v_1, \ldots, v_n)$ be the n vertices of P; assume they are in general position. Let R be the set of reflex vertices of P. For every reflex vertex $v \in R$, the angle bisector of the interior angle at v hits some edge $a_v b_v$ of P. Let $L = \{v, a_v, b_v : v \in R\}$, that is, L is the set of all reflex vertices of P, and both endpoints of the edges hit by the angle bisectors of reflex angles. Clearly, $|L| \leq 3r$.

▶ **Lemma 6.** *There is a simple polygon $Q \subset P$ whose vertex set is L, and every connected component of $P \setminus Q$ is a convex polygon. The polygon Q can be computed in $O(n \log n)$ time.*

Proof. We describe an algorithm that decomposes P along noncrossing diagonals into a collection $\mathcal{P}$ of convex polygons and their complement $P \setminus (\bigcup_{A\in\mathcal{P}} A)$ will be polygon Q. Initially $\mathcal{P} = \{P\}$ and $Q = \emptyset$.

The algorithm has two steps. In the first step, a collection $\mathcal{P}$ of convex polygons is created such that the vertex set of the complement $P \setminus (\bigcup_{A\in\mathcal{P}} A)$ is L. However, $P \setminus (\bigcup_{A\in\mathcal{P}} A)$ is not necessarily connected. In the second step, the connected components of $P \setminus (\bigcup_{A\in\mathcal{P}} A)$ are merged into a simple polygon Q (a single connected component) with the same vertex set L.

First step. While there is a nonconvex polygon $P' \in \mathcal{P}$, we replace P' with one or more smaller polygons in $\mathcal{P}$ as follows. Let v be a reflex vertex of P'. Since $P' \subset P$, vertex v is also a reflex vertex of P. Denote by $\vec{r}_v$ the angle bisector of P at v. Note that $\vec{r}_v$ enters the interior of P' at v; denote by ab the edge of P' where $\vec{r}_v$ first exits P'. Let $P(v)$ be the geodesic triangle formed by the edge ab and the shortest paths from v to a and to b, respectively. Update $\mathcal{P}$ by replacing P' with the polygons in $P' \setminus P(v)$. See Figure 4 for an example. In the course of the algorithm, every polygon in $\mathcal{P}$ is formed by a sequence of consecutive vertices of the input polygon P.

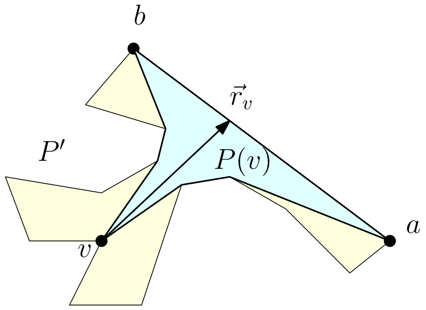

Figure 4 Replace P' by four polygons (in yellow), after taking out the geodesic triangle $P(v)$ where v is a reflex vertex.

We claim that in each iteration of the algorithm, all vertices of $P(v)$ are in L. Clearly, v is a reflex vertex in P', hence a reflex vertex of P, as well. Similarly, the interior vertices of the shortest paths from v to a and to b are reflex vertices in P', hence in P. It remains to show that $a, b \in L$. If ab is an edge of P, then $a, b \in L$ by the definition of L. Otherwise, ab is a diagonal of P, and so it is an edge of a geodesic triangle $P(v')$ of some previous iteration of the algorithm – by induction, they are in L, as well. At the end of the while loop, all polygons in $\mathcal{P}$ are convex, however, the complement $P \setminus (\bigcup_{A\in\mathcal{P}} A)$ is not necessarily connected. See Figure 5 for an illustration.

Second step. While there is a (convex) polygon $P' \in \mathcal{P}$ incident to three or more vertices in L, we replace P' with smaller polygons: In particular, let $V(P')$ be the vertex set of P'. If $|V(P') \cap L| \geq 3$, then replace P' with the polygons in $P' \setminus \text{conv}(V(P') \cap L)$, where conv(.) stands for the convex hull. In each iteration, all polygons in $\mathcal{P}$ remain convex. At the end of the while loop, every polygon in $\mathcal{P}$ is incident to exactly two vertices in L, and $P \setminus (\bigcup_{A\in\mathcal{P}} A)$ is a simple polygon with vertex set L. ◀

▶ **Lemma 7.** *Every simple polygon P on n vertices, r of which are reflex, can be decomposed into convex faces such that every chord of P intersects $O(\log r)$ faces. Such a decomposition can be computed in $O(n \log n)$ time.*

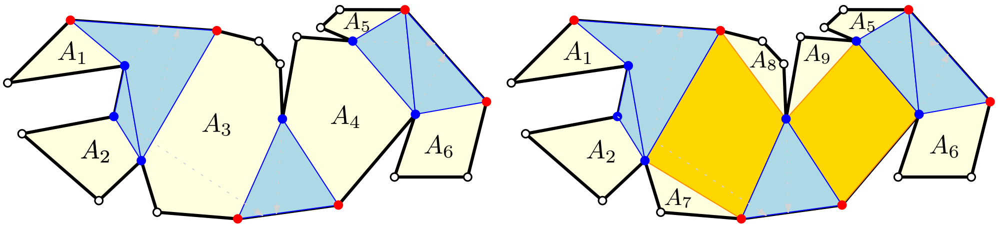

Figure 5 A simple polygon P, the vertices in L are blue (reflex) or red (hit by angle bisector). Left: The first step produces convex polygons $\mathcal{P} = \{A_1, \ldots, A_6\}$, but $P \setminus \bigcup_{i=1}^{6} A_i$ is disconnected. Right: The second step merges $P \setminus \bigcup_{i=1}^{6} A_i$ into a simple polygon Q. As $|A_3 \cap L| \geq 3$ and $|A_4 \cap L| \geq 3$, the second step creates $\text{conv}(V(A_3) \cap L)$ and $\text{conv}(V(A_4) \cap L)$ (shown in deep yellow) which merges $P \setminus \bigcup_{i=1}^{6} A_i$ into a single connected component Q.

Proof. We can compute the set L of up to $3r$ vertices and a simple polygon $Q \subset P$ described in Lemma 6 in $O(n \log n)$ time. We then compute the Hershberger–Suri triangulation for Q, which is a Steiner triangulation of $O(r)$ triangles such that every chord of Q intersects $O(\log r)$ triangles [17]. This triangulation of Q, together with the convex polygons in $P \setminus Q$, form a subdivision of P into convex faces.

We claim that every chord of P intersects at most $O(\log r)$ faces: at most $O(\log r)$ triangles in Q and at most two convex sets in $P \setminus Q$. If a chord ℓ of P intersects three components of $P \setminus Q$, say C_1, C_2, C_3 in this order, then ℓ crosses the boundary of C_2 twice, so C_2 must have at least two edges on the boundary between C_2 and Q. However, by Lemma 6, every edge of Q is either an edge or a diagonal of P. Therefore the boundary between Q and a component of $P \setminus Q$ is a single diagonal of P. Thus ℓ intersects at most two components of $P \setminus Q$; moreover $\ell \cap Q$ is a chord of Q, so it intersects $O(\log r)$ triangles inside Q. ◀

By performing the algorithm on the convex subdivision in Corollary 7, the approximation ratio improves to $O(\log r)$.

▶ **Theorem 8.** *Let P be a simple polygon with n vertices, r of which are reflex, and let k^* be the minimum number of lasers needed to subdivide P into pieces of area at most 1. We can find an integer k with $k^* \leq k \leq O(k^* \log r)$ in $O(n \log n)$ time, and a set of k lasers that subdivide P into pieces of area at most 1 in output-sensitive $O(k + n \log n)$ time.*

3.2 Bi-Criteria Approximation for Diameter

For the diameter version in a simple polygon, we describe a bi-criteria approximation algorithm (Theorem 11). We start from deriving a lower bound for the minimum number of lasers in a decomposition into pieces of diameter at most δ (for bi-criteria approximation algorithm we use general δ, instead of δ =1, because we will scroll over δ when using the algorithm to get an approximation for MinDiameter).

Consider the infinite set of vertical lines, $\mathcal{L}_V$, evenly spaced with separation δ; that is, $\mathcal{L}_V = \{x = i\delta : i \in \mathbb{Z}\}$. The lines in $\mathcal{L}_V$ decompose P into a set $\mathcal{P}_V$ of simple polygons, that we call *cells*. By construction, the orthogonal projection of each cell to the x-axis is an interval of length at most δ. (More precisely, we consider the polygon P to be a closed set in the plane. Subtracting the union of vertical lines $\mathcal{L}_V$ from P results in a set of connected components; the closures of these components are the simple polygons in $\mathcal{P}_V$). The polygons in $\mathcal{P}_V$ are faces in the arrangement of the lines in $\mathcal{L}_V$ and the edges of P; the planar dual of this decomposition is a tree, whose nodes are the faces $\mathcal{P}_V$ and whose edges are dual to the vertical lines.

If the projection of polygon $Q \in \mathcal{P}_V$ onto the x-axis is an interval of length δ (which means it extends from $x = i\delta$ to $x = (i+1)\delta$, for some integer i), we say that Q is a *full-width* cell; otherwise the projection of Q onto the x-axis is of length less than δ, and we say that Q is a *narrow* cell. (It may be that P itself is a narrow cell if, e.g., P does not intersect any of the vertical lines L_V.)

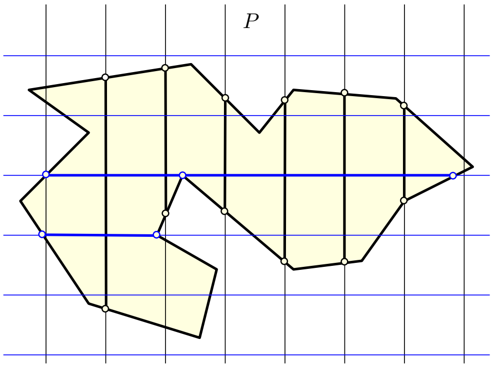

Figure 6 P is subdivided by a grid; the lasers are thick.

The intersection of the lines in $\mathcal{L}_V$ with P is a set of vertical chords of P. Let C_V be the set of these chords. While there is a chord $\ell \in C_V$ that lies on the boundary of some narrow cell, remove ℓ from C_V (thereby merging the cells on the two sides of ℓ into one cell). As a result, all remaining chords lie on the boundary between full-width cells. Let C'_V be the resulting set of chords, and let $k_V = |C'_V|$ denote their cardinality. Since any two full-width cells of $\mathcal{P}_V$ that are in adjacent vertical strips remain separated by a chord in C'_V, the x-extent of each face in the new decomposition of P is at most 3δ. We summarize this below.

▶ **Proposition 9.** *The remaining k_V chords C'_V, $k_V \geq 0$, subdivide P into a set $\mathcal{Q}$ of $k_V + 1$ polygons, each of which intersects at most two lines in $\mathcal{L}_V$, consequently its projection to the x-axis is an interval of length less than 3δ. Further, the dual graph of this decomposition is a tree (with k_V edges and $k_V + 1$ nodes).*

If $k_V = 0$, then there is just one cell, $\mathcal{Q} = \{P\}$. If $k_V \geq 1$, then each $Q \in \mathcal{Q}$ includes at least one full-width cell, since the only lasers remaining are those separating one full-width cell from an adjacent full-width cell sharing the laser.

Thus, the boundary of each $Q \in \mathcal{Q}$ includes at least two distinct (simple) paths connecting a point on one line of $\mathcal{L}_V$ to a point on an adjacent line of $\mathcal{L}_V$. Each of these paths has length at least δ. The endpoints of such a path are at distance at least δ away from each other. In any laser cutting of P into pieces of diameter at most δ, each such path contains a laser endpoint in its interior or at both endpoints (if the path is a horizontal line segment). In any case, each of these paths contains a laser endpoint in its interior or at its left endpoint. Thus overall, there must be at least $2|\mathcal{Q}| = 2(k_V + 1)$ endpoints of lasers. This implies that $k^* \geq k_V + 1$, where k^* is the minimum number of lasers in order to achieve pieces of diameter at most δ. Therefore we conclude,

▶ **Lemma 10.** *If $k_V \geq 1$, then $k^* \geq k_V + 1$.*

Now, we consider the set of horizontal lines, $\mathcal{L}_H = \{y = j\delta : j \in \mathbb{Z}\}$, and apply the above process to polygon P, yielding horizontal chords C_H, and then a subset $C'_H \subseteq C_H$ of chords after merging cells (removing lasers that separate a *full-height* cell from an adjacent *short* cell). The result is a decomposition of P into $k_H + 1 = |C'_H| + 1$ pieces, each having projection onto the y-axis of length less than 3δ. Analogously to Lemma 10, we get $k^* \geq k_H + 1$ if $k_H \geq 1$.

If we now overlay the vertical chords C'_V and the horizontal chords C'_H, the resulting arrangement decomposes P into pieces each of which is a simple polygon having projections onto both the x- and the y-axis of lengths less than 3δ; thus, the resulting pieces each have diameter less than $3\delta\sqrt{2}$. The total number of lasers is $k_V + k_H \leq 2(k^* - 1)$.

▶ **Theorem 11.** *Let P be a simple polygon with n vertices, and let k^* be the minimum number of lasers that decompose P into pieces each of diameter at most δ for a fixed $\delta > 0$. One can compute a set of at most $2(k^* - 1)$ axis-aligned lasers that decompose P into pieces each of diameter at most $3\sqrt{2}\delta$ in time polynomial in n and $\mathrm{diam}(P)/\delta$.*

3.3 $O(1)$-Approximation for MinDiameter in Simple Polygons

In this section we consider the problem of minimizing the maximum diameter of a cell in the arrangement of k lasers, for a given number k. Our $O(1)$-approximation algorithm repeatedly decreases the x- and y- separation in the bi-criteria solution from Theorem 11 until the number of placed lasers is about to jump over $2k$; then, the number of lasers is halved while increasing the diameter by a constant factor.

Specifically, let $\ell(\delta)$ denote the number of lasers used in the end of the bi-criteria algorithm with the x- and y-separation between consecutive vertical and horizontal lines being δ. Our algorithm to approximate the diameter achievable with k lasers is as follows:

- Initialize $\delta = \mathrm{diam}(P)$, and $\varepsilon > 0$.
- While $\ell(\delta) \leq 2k$, set $\delta = \delta/(1+\varepsilon)$ and recompute $\ell(\delta)$.
- Let δ_0 be such that $\ell(\delta_0) \leq 2k$ but $\ell(\delta_0/(1+\varepsilon)) > 2k$.
- Let C_V and C_H be the $\ell(\delta_0) \leq 2k$ vertical and horizontal lasers, resp., found by the bi-criteria algorithm.
- Partition C_V into lasers along $x = i\delta_0$ for even i and the rest (odd i); let C'_V be a smallest part. Similarly, let C_H be a smaller part when we partition C_H into two subsets of lines where $y = i\delta_0$ is an even or odd multiple of δ_0.
- Return the lasers in $C'_V \cup C'_H$.

▶ **Theorem 12.** *Let P be a simple polygon with n vertices, and let δ^* be the optimal diameter achievable with k lasers. For every $\varepsilon > 0$, one can compute a set of at most k lasers that partition P into pieces each of diameter at most $4\sqrt{2}(1+\varepsilon)\delta^*$ in time polynomial in n, $\mathrm{diam}(P)/\delta^*$, and ε.*

The proof of Theorem 12 is presented in the full version of this paper.

4 Axis-Parallel Lasers

In this section we study Min-LaserDiameter and Min-LaserArea under the constraint that all lasers must be axis-parallel (the edges of P may have arbitrary orientations). The algorithms for both problems start with a "window partitioning" P into "(pseudo-)histograms" of stabbing number at most three, and are then tuned to the specific measures to partition the histograms. We use a simple sweepline algorithm for the diameter, and a dynamic program for the area. The main result is:

▶ **Theorem 13.** *Let P be a simple polygon with n vertices and let k^* be the minimum number of axis-parallel lasers needed to subdivide P into pieces of area (diameter) at most 1. There is an algorithm that finds $O(k^*)$ axis-parallel lasers that subdivide P into pieces of area (diameter) at most 1 in time polynomial in n and $\mathrm{area}(P)$ $(\mathrm{diam}(P))$.*

4.1 Reduction to Histograms

A *histogram* is a simple polygon bounded by an axis-parallel line segment, the *base*, and an x- or y-monotone polygonal chain between the endpoints of the base.

The *window partition* of a simple polygon was originally used for the design of data structures that support link distance queries [24, 30]. In this section, we use the axis-parallel version, which partitions a simple polygon P into histograms such that every axis-parallel chord of P intersects at most 3 histograms. Window partitions for orthogonal polygons can be computed by a standard recursion [24, 30]; we use a modified version where we recurse until the remaining subpolygons are below the size threshold 1. This modification guarantees termination on all simple polygons (not only orthogonal polygons).

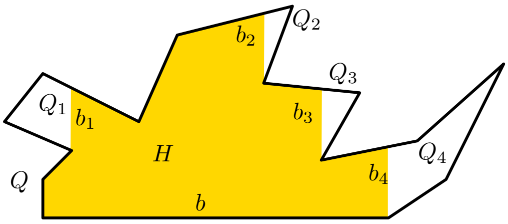

Figure 7 Window partition of a polygon Q with a horizontal base b into a maximal histogram H (colored gold) and four polygons Q_1, Q_2, Q_3, and Q_4 (in white). If the sizes (areas/diameters) of Q_1, Q_2, Q_3, and Q_4 are each at most 1, then Q is a pseudo-histogram.

Window Partition Algorithm. Given a simple polygon P, let b be an axis-parallel chord of P that subdivides P into two simple polygons P_1 and P_2 with a common base b. Let $\mathcal{S} = \{(P_1, b), (P_2, b)\}$ be a set of tuples where each tuple has a polygon and its axis-parallel base, and let $\mathcal{H} = \emptyset$ be the set of histograms. While $\mathcal{S}$ contains a pair (Q, b), where the size (e.g., the diameter) of Q is more than 1, do the following:

1. compute the maximal histogram[1] H of base b in Q, and add (H, b) to $\mathcal{H}$; see Figure 7;
2. update $\mathcal{S}$ by replacing (Q, b) with the pairs (Q_i, b_i), where the polygons Q_i are the connected components of $Q \setminus H$, and b_i is the boundary between Q_i and H.

Return $\mathcal{H}$ and $\mathcal{S}$.

Pseudo-histograms. Let T_1 and T_2 be the recursion trees of the algorithm, rooted at P_1 and P_2, respectively. Let $T = T_1 \cup T_2$. Each node $v \in T$ corresponds to a polygon $Q_v \subset P$. Every nonleaf node $v \in T$ also corresponds to a histogram $H_v \subset Q_v$; it is possible that size$(H_v) \leq 1$ but size$(Q_v) > 1$ (the size is area or diam based on the problem). For a leaf $v \in T$, we have either size$(Q_v) \leq 1$, or $H_v = Q_v$ and size$(H_v) > 1$. The polygons Q_v at leaf nodes and the histograms H_v at nonleaf nodes jointly form a subdivision of P.

Every node $v \in T$ in the recursion tree corresponds to a polygon-base pair (Q_v, b_v). For any subset $U \subset V(T)$, where $V(T)$ is the set of vertices of T, the bases $\{b_u : u \in U\}$ decompose P into simply connected cells. For every $u \in U$, there is a cell P_u in the decomposition such that $H_u \subset P_u \subset Q_u$. Since every axis-parallel chord of P crosses at most 2 bases, it can intersect at most 3 polygons in such a decomposition.

[1] Without loss of generality, assume b is horizontal. H can be obtained by taking all points of Q reachable through a vertical line from points on b.

In a bottom-up traversal of T, we can find a subset $U \subset V(T)$ such that $\{b_u : u \in U\}$ decomposes P into polygons P_u, $u \in U$, such that size$(P_u) > 1$ but the size of every component of $P_u \setminus H_u$ is at most 1. Each polygon P_u consists of a histogram H_u with base b_u, and subpolygons (*pockets*) of size at most 1 attached to some edges of H_u orthogonal to b_u. We call each such polygon P_u a *pseudo-histogram*. See Figure 7.

4.2 $O(1)$-Approximation for Min-LaserDiameter in Histograms

We start with an $O(1)$-approximation for histograms, and then extend our algorithm to pseudo-histograms and simple polygons. Without loss of generality, we assume that the base is horizontal.

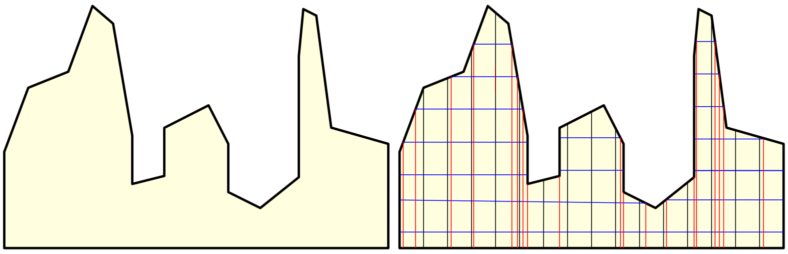

Figure 8 Left: A histogram polygon with a horizontal base. Right: lasers introduced in Phase 1 are shown in black. Horizontal (vertical) lasers introduced in Phase 2 are shown in blue (red).

▶ **Theorem 14.** *There is an algorithm that, given a histogram P with n vertices, computes an $O(1)$-approximation for the axis-parallel* Min-LaserDiameter *problem in time polynomial in n and* diam(P)*.*

Proof. We first describe the algorithm.

Algorithm. We are given a histogram P with a horizontal base ab. If diam$(P) \le 1$, halt. Otherwise, do the following:

1. Subdivide ab into $\lceil 2|ab| \rceil$ intervals which all, except possibly one, have length 1/2 and place vertical lasers on the boundary between consecutive intervals.
2. Sweep P with a horizontal line L top down, and maintain the set of cells formed by all lasers in step one and the line L. When the diameter of a cell C above L is precisely 1, place a horizontal laser pq along the bottom side of C, where $p, q \in \partial P$, and place two vertical lasers at p and q, respectively.

Analysis. Let OPT denote the set of lasers in an optimal solution, and let $k^* = |\text{OPT}|$. Denote by ALG the set of lasers computed by the algorithm; let ALG^1 be the number of vertical lasers computed in Phase 1, and let ALG^2_h and ALG^2_v be the set of horizontal and vertical lasers computed in Phase 2. Clearly, $|\text{ALG}^2_v| \le 2|\text{ALG}^2_h|$. Therefore it is enough to prove that $|\text{ALG}^1| = O(k^*)$ and $|\text{ALG}^2_h| = O(k^*)$.

First we show that $|\text{ALG}^1| = O(k^*)$. The vertical lasers in OPT subdivide the base ab into segments of length at most 1. Therefore, $k^* \ge \lfloor |ab| \rfloor$. Combined with $k^* \ge 1$, this readily implies that $|\text{ALG}^1| = \lceil 2|ab| \rceil - 1 = O(k^*)$.

Next we prove that $|\text{ALG}^2_h| \le 2k^*$ using a charging scheme. Specifically, we charge every laser in ALG^2_h to a laser in OPT such that each laser in OPT is charged at most twice. Recall for each laser $pq \in \text{ALG}^2_h$ placed by the algorithm, there is a cell $C = C_{pq}$ such that diam$(C_{pq}) = 1$ and pq contains the bottom edge of C_{pq}. Since diam$(C_{pq}) = 1$, the cell C_{pq}

intersects some laser in OPT; we charge pq to one of these lasers. Denote by $\mathrm{OPT}_h(C_{pq})$ and $\mathrm{OPT}_v(C_{pq})$, resp., the horizontal and vertical lasers in OPT that intersect C_{pq}. The charging scheme is described by the following rules:

(a) If $\mathrm{OPT}_h(C_{pq}) \neq \emptyset$, then charge pq to the lowest laser in $\mathrm{OPT}_h(C_{pq})$;
(b) else, if C_{pq} intersects ∂P, then charge pq to a laser in $\mathrm{OPT}_v(C_{pq})$ that is closest to $C_{pq} \cap \partial P$;
(c) else, if there is no horizontal laser in OPT that lies above pq, then charge pq to an arbitrary laser in $\mathrm{OPT}_v(C_{pq})$;
(d) else, charge pq to the lowest horizontal laser in OPT that lies above pq.

It remains to prove that each laser in OPT is charged at most once for each the four rules. Since (a) and (d) charge to horizontal lasers, and (b) and (c) charge to vertical lasers in OPT, then each laser in OPT is charged by at most two of the rules. In each case, we argue by contradiction. Assume that a laser $\ell \in \mathrm{OPT}$ is charged twice by one of the rules, that is, there are two lasers $pq, rs \in \mathrm{ALG}_h^2$, that are charged to ℓ by the same rule. The width of cells C_{pq} and C_{rs} is at most $1/2$, because of the spacing of the vertical lasers in ALG^1. Since $\mathrm{diam}(C_{pq}) = \mathrm{diam}(C_{rs}) = 1$, they each have height at least $\sqrt{3}/2$. Without loss of generality, we may assume that the algorithm chooses pq before rs.

(a) In this case, ℓ is the lowest horizontal laser in OPT that intersect C_{pq} and C_{rs}, respectively. Since pq is above rs, laser pq intersects the interior of C_{rs}, contradicting the assumption that C_{rs} is a cell formed by the arrangement of all lasers in ALG.
(b) In this case, ℓ is a vertical laser that intersects both C_{pq} and C_{rs}, and also intersect ∂P. When the algorithm places a horizontal laser at pq, it also places vertical lasers from p and q to the base of P. These three lasers separate ∂P from the portion of ℓ below pq. This contradict the assumption that C_{rs} is a cell formed by the arrangement of all lasers in ALG.
(c) In this case, both C_{pq} and C_{rs} intersects a vertical laser $\ell \in \mathrm{OPT}$, and they both lie above the highest horizontal laser that crosses ℓ. Consequently, they both intersect the two highest cells, say C_{left} and C_{right}, on the two sides of ℓ in the arrangement formed by OPT. The combined height of C_{pq} and C_{rs} is at least $\sqrt{3}$. Therefore, the height of C_{left} and C_{right} is at least $\sqrt{3} > 1$, contradicting the assumption that $\mathrm{diam}(C_{\text{left}}) \leq 1$ and $\mathrm{diam}(C_{\text{right}}) \leq 1$.
(d) In this case, ℓ is the lowest horizontal laser in OPT that lies above C_{pq} and C_{rs}, respectively. Let C_{below} be the cell of the arrangement of OPT that lies below ℓ. The combined height of C_{pq} and C_{rs} is at least $\sqrt{3}$. Therefore, the height of C_{below} is at least $\sqrt{3} > 1$, contradicting the assumption that $\mathrm{diam}(C_{\text{below}}) \leq 1$. ◀

Adaptation to Pseudo-Histograms. In a laser cutting of P into pieces of diameter at most 1, each pseudo-histogram P_u intersects a laser, since $\mathrm{diam}(P_u) > 1$. An adaptation of the algorithm in Section 4.2 can find an $O(1)$-approximation for Min-LaserDiameter in each P_u. As noted above, each laser intersect at most 3 pseudo-histograms, hence the union of lasers in the solutions for pseudo-histograms is an $O(1)$-approximations for P.

The sweepline algorithm in Section 4.2 can easily be adapted to subdivide a pseudo-histogram P_u. Recall that P_u consists of a histogram H_u and pockets of diameter at most 1. We run steps 1 and 2 of the algorithm for the histogram H_u with two minor changes in step 2: (1) we compute the *critical* diameters with respect to P_u (rather than H_u), and (2) when the diameter of a cell C above a chord pq of P_u is precisely 1, we place up to *four* vertical lasers: at intersection points of L with ∂P_u the ∂H_u (the vertical lasers through $pq \cap \partial H_u$ cut possible pockets that intersect pq). The analysis of the sweepline algorithm is analogous to Section 4.2, and yields the following result.

▶ **Theorem 15.** *There is an algorithm that, given a simple polygon P with n vertices,computes an $O(1)$-approximation for the axis-parallel* Min-LaserDiameter *problem in time polynomial in n and* $\mathrm{diam}(P)$.

4.3 Discretization of the Solution Space in a Histogram Polygon

Consider a histogram polygon P having n vertices. We assume that the vertices are in general position, in the sense that no three vertices are collinear. We show that there is a discrete set of candidate orthogonal chords such that lasers chosen from this set yield an $O(1)$ approximation for minimizing the number of lasers, subject to a target measure bound 1 on the obtained pieces. This is useful for finding an $O(1)$ approximation for Min-LaserArea by dynamic programming.

We prove the following lemma in the full version, which extends to pseudo-histograms.

▶ **Lemma 16.** *For a histogram P, let k^* be the minimum number of axis-parallel lasers that subdivide P into pieces of area (diameter) at most 1. We can find a set C of $O(n + \mathrm{area}(P))$ $(O(n + per(P)))$ chords of P, such that $O(k^*)$ lasers from C can subdivide P into pieces of area (diameter) at most 1.*

4.4 $O(1)$-Approximation for Min-LaserArea

We now consider the Min-LaserArea with axis-parallel lasers chosen from a discrete set to achieve pieces of area at most 1. An $O(1)$-approximation algorithm is based on the window partition method described earlier, allowing us to reduce to the case of subdividing a pseudo-histogram, for which we give a dynamic program to choose lasers from the discrete candidate set. In the full version, we describe the algorithm using area as the measure. With slight modifications, the algorithm also applies to the measure of diameter, allowing us to solve Min-LaserDiameter in pseudo-histograms (albeit in a higher polynomial time bound than stated in Theorem 14).

5 Diameter Measure in Polygons with Holes and Axis-Parallel Lasers

5.1 Bi-Criteria Approximation for Diameter

In this section we give a bi-criteria approximation for the diameter version in a polygon with holes when lasers are constrained to be axis-parallel. The approach is similar to the algorithm for simple polygons and lasers of arbitrary orientations (cf. Section 3.2) in that both use grid lines, but they differ significantly to handle holes in a polygon when the lasers are axis-parallel. Particularly, in simple polygons we place lasers along grid lines, while in polygons with holes the grid lines just divide the problem into sub-problems.

Lasers in Vertical Strips. Consider the infinite set of equally spaced vertical lines $\mathcal{L}_V = \{x = i\delta : i \in \mathbb{Z}\}$, for some $\delta > 0$. The lines subdivide P into a set $\mathcal{P}_V$ of polygons (possibly with holes), that we call *strips*. (Unlike Section 3.2, we do not place lasers along the lines in $\mathcal{L}_V$; we use the strips for a divide-and-conquer strategy.) The projection of any strip on the x-axis has length at most δ; we say that a strip is *full-width* if its projection has length exactly δ. Let $\mathcal{F}_V \subset \mathcal{P}_V$ denote the set of full-width strips, and let $F \in \mathcal{F}_V$ be a full-width strip.

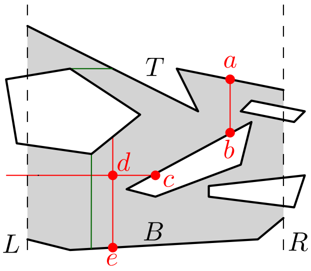

Figure 9 F is shaded, the holes are white. The L-R separating path $\gamma = abcde$ (vertices marked with red disks) alternates between holes and lasers (red) in the interior of P; $a_F(\gamma) = 2$ as there are two links ab and de in path γ whose extensions are fully contained in F. ab and cde are the maximal rectilinear subpaths of γ through the free space. The minimum-link path $\gamma(F)$ (darkgreen) also alternates between holes and free space.

The leftmost (resp., rightmost) points of F lie on a line $L = \{x = i\delta\}$ (resp., a line $R = \{x = (i+1)\delta\}$) for some $i \in \mathbb{Z}$ (see Fig. 9). Consequently, the outer boundary of F contains two simple paths between L and R; we denote them by T (top) and B (bottom).

Since the distance between L and R is δ, in any laser cutting of P into cells of diameter at most δ, there exists a T-B path $\gamma \subset F$ along the boundaries of cells that separates L and R. Since γ is disjoint from the interior of the cells, it must follow lasers in the interior of P. We may assume, w.l.o.g., that γ follows any laser at most once; otherwise we could shortcut γ along the laser. Since the lasers are axis-aligned, γ is an alternating sequence of subpaths that are either in ∂P or rectilinear paths through the interior of F; we call any such T-B path an *alternating* path.

An axis-aligned segment s, fully contained in F, is *associated* with F if it remains fully contained in F after it is maximally extended within P (i.e., if both endpoints of the supporting chord of P are on the boundary of F). For example, any *vertical* segment $s \subset F$ is associated with F (because T and B belong to the boundary of F). Let $a_F(\gamma)$ be the number of associated links of γ (i.e., the number of edges whose supporting chords are fully contained in F). Let $|\gamma|$ denote the total number of the (axis-aligned) edges in γ. A key observation is the following.

▶ **Lemma 17.** $|\gamma| \leq 3\, a_F(\gamma)$.

Proof. Let π be a (maximal) rectilinear subpath of γ through the free space, i.e., a part of γ whose endpoints lie on the boundary of P. If π is a single horizontal link, then the link is associated with F (because if any of its two ednpoints is outside F, then γ protrudes through L or R, not separating them). Otherwise (i.e., if π contains vertical links), the number of the vertical links is at least 1/3 of the total number of links in π. The lemma follows by summation over all subpaths of γ. ◀

Our algorithm computes an alternating path $\gamma(F)$ with the minimum number of links and places one laser along every link of $\gamma(F)$ (the horizontal lasers may extend beyond F). To find $\gamma(F)$, we can build the *critical graph* of F, whose vertices are T, B, and components of ∂P within the strip F (including holes in the strip), and in which the weight of the edge between two vertices is the axis-parallel link distance between them. The weight of an edge between vertices i and j can be found by in polynomial time by standard wave propagation techniques [10, 26], i.e., by successively labeling the areas reachable with k links from i for increasing k, until j is hit by the wave. After the critical graph is built, $\gamma(F)$ is found as the shortest T-B path in the graph.

By minimality of $\gamma(F)$, the number links $|\gamma(F)|$ in it (and hence the number of lasers we place) is at most $|\gamma(F)|$. Let $k_V = \sum_{F\in\mathcal{F}_V} |\gamma(F)|$ be the total number of lasers placed in all full-width strips in $\mathcal{F}_V$, and let k^* be the minimum number of axis-parallel lasers in a laser cutting of P into cells of diameter at most δ. An immediate consequence of Lemma 17 is the following.

▶ **Corollary 18.** *$k_V \leq 3k^*$.*

Proof. As the links of $\gamma(F)$ follow lasers, at least $a_F(\gamma)$ lasers are fully contained in F. ◀

The k_V lasers placed in full-width strips subdivide P into polygonal pieces; let Q be one such piece.

▶ **Lemma 19.** *The length of the x-projection of Q on the x-axis is at most 2δ.*

Proof. We prove that Q intersects at most one line in $\mathcal{L}_V$. Suppose, to the contrary, that Q intersects two consecutive lines $\ell_1 : x = i\delta$ and $\ell_2 = x = (i+1)\delta$. Let λ be a shortest path in Q between points in $Q \cap \ell_1$ and $Q \cap \ell_2$, respectively. By minimality, λ lies in the strip between ℓ_1 and ℓ_2. Consequently, λ is contained in some full-width strip $F \subset \mathcal{F}_V$. However, the path $\gamma(F)$ intersects every path in F between $F \cap \ell_1$ and $F \cap \ell_2$; in particular, it intersects λ. Since we have placed a laser along every segment of $\gamma(F)$ in the interior of P, λ intersects a laser, contradicting the assumption that $\lambda \subset Q$. ◀

Lasers in Horizontal Strips. Similarly, we consider the set of horizontal lines $\mathcal{L}_H = \{y = j\delta : j \in \mathbb{Z}\}$ and apply the above process to P, yielding horizontal chords C_H that subdivide the polygon into horizontal strips (polygons, possibly with holes). We again work only with *full-height* strips, whose boundary intersect two consecutive lines in $\mathcal{L}_H$. In each full-height strip, we find a minimum-interior-link rectilinear path that separates the boundary points along the two lines in $\mathcal{L}_H$, and place lasers along the links of the path. Let k_H be the number of lasers over all full-height strips.

Putting Everything Together. We overlay the k_V lasers in full-width strips with the k_H lasers in full-height strips. The resulting arrangement partitions P into polygonal pieces (possibly with holes). The x- and y-projection of each piece has length at most 2δ by Lemma 19; thus, each piece has diameter less than $2\delta\sqrt{2}$. By Corollary 18, the total number of lasers used in the arrangement is $k_V + k_H \leq 6k^*$. We obtain the following theorem.

▶ **Theorem 20.** *Let P be a polygon with holes of diameter $\operatorname{diam}(P)$ having n vertices, and let k^* be the minimum number of laser cuts that partition P into pieces each of diameter at most δ for a fixed $\delta > 0$. In time polynomial in n and $\operatorname{diam}(P)/\delta$, one can compute a set of at most $6k^*$ lasers that subdivide P into pieces each of diameter at most $2^{3/2}\delta$.*

5.2 $O(1)$-Approximation to MinDiameter

Similarly to Section 3.3, we can use the bi-criteria algorithm to derive a constant-factor approximation for minimizing the maximum diameter of a cell in the arrangement of a given number k of axis-parallel lasers. Our $O(1)$-approximation algorithm (Theorem 21, proof in the full version) repeatedly decreases the x- and y- separation in the bi-criteria solution from Theorem 20 until the number of placed lasers is about to jump over $6k$; then, the number of lasers is decreased by a factor of 6 while increasing the diameter by a constant factor.

▶ **Theorem 21.** *Let δ^* be the minimum diameter achievable with k axis-parallel lasers. For every $\varepsilon > 0$, one can compute a set of at most k axis-parallel lasers that partition P into pieces each of diameter at most $12\sqrt{2}(1+\varepsilon)\delta^*$ in time polynomial in n, $\operatorname{diam}(P)/\delta^*$, and ε.*

6 $O(\log \mathrm{OPT})$-approximation for Min-LaserCircle

This section considers the radius of the largest inscribed circle as the measure of cell size; in particular, in Min-LaserCircle the goal is to split the polygon P (which may have holes) into pieces so that no piece contains a disk of radius 1. We give an $O(\log \mathrm{OPT})$-approximation algorithm for Min-LaserCircle based on reducing the problem to SetCover. The following reformulation is crucial for the approximation algorithm:

▶ **Observation 22.** *A set of lasers splits P into pieces of in-circle radius at most 1 if and only if every unit disk that lies inside P is hit by a laser.*

▶ **Theorem 23.** *For a polygon P with n vertices (possibly with holes), Min-LaserCircle admits an $O(\log \mathrm{OPT})$-approximation in time polynomial in n and $\mathrm{area}(P)$.*

Proof. We lay out a regular square grid of points at spacing of $\sqrt{2}$. The set of grid points within P is denoted by G. We may assume $|G| = O(\mathrm{area}(P))$ by a suitable (e.g., uniformly random) shift. Due to the spacing, every unit-radius disk in P contains a point of G (possibly on its boundary).

Consider an optimal set L^* of lasers that hit all unit disks that are contained within P. Replace each laser (chord) $c \in L^*$ with up to four anchored chords of the same homotopy type as c with respect to the vertices of P and the points G, obtained as follows: Shift the chord c vertically down (up), while keeping its endpoints on the same pair of edges of P, until it becomes incident to a point in G or a vertex of P, then rotate the chord clockwise (counterclockwise) around this point until it becomes incident to another point in G or a vertex of P. Since every unit disk within P contains a point of G, any unit disk within P that is intersected by c is also intersected by one of the shifted and rotated copies of c. This means that we can construct a candidate set, C, of $O((n + \mathrm{area}(P))^2)$ chords that can serve as lasers in an approximate solution, giving up at most a factor 4 of optimal. Further, in the arrangement of the segments C within P, any unit disk is intersected by some set of chords of C, thereby defining a combinatorial type for each unit disk in P. (Two disks are of the same type if they are intersected by the same subset of chords in C; one way to define the type is to associate it with a cell in the arrangement of lines drawn parallel to each chord $c \in C$ at distance 2 from c on each side of c. While the center of the disk is in one cell of the arrangement, the disk intersects the same chords.) Let D be the polynomial-size ($O(|C|^2)$) set of disks, one "pinned" (by two segments, from the set C and the set of edges of P) disk per combinatorial type. By construction, any set of chords from C that meets all disks of D must meet all unit disks within P.

We thus formulate a discrete set cover instance in which the "sets" correspond to the candidate set C of chords, and the "elements" being covered are the disks D. Since there are constant-size sets of disks that cannot be shattered, the VC dimension of the set system is constant, and an $O(\log \mathrm{OPT})$-approximate solution for the set cover can be found in time polynomial in the size of the instance [7]. ◀

The same algorithm works for the version in which the lasers are restricted to be axis-aligned (the only change is that the candidate set C consists of axis-aligned chords through points of the grid G and vertices of P).

References

1 Bogdan Armaselu and Ovidiu Daescu. Algorithms for fair partitioning of convex polygons. *Theoretical Computer Science*, 607:351–362, 2015. doi:10.1016/j.tcs.2015.08.003.

2 Imre Bárány, Pavle Blagojević, and András Szűcs. Equipartitioning by a convex 3-fan. *Advances in Mathematics*, 223(2):579–593, 2010. doi:10.1016/j.aim.2009.08.016.

3 Aandrás Bezdek and Károly Bezdek. A solution of Conway's fried potato problem. *Bulletin of the London Mathematical Society*, 27(5):492–496, 1995. doi:10.1112/blms/27.5.492.

4 Pavle V. M. Blagojević and Günter M. Ziegler. Convex equipartitions via equivariant obstruction theory. *Israel Journal of Mathematics*, 200(1):49–77, 2014. doi:10.1007/s11856-014-1006-6.

5 Karol Borsuk. Drei Sätze über die n-dimensionale euklidische Sphäre. *Fundamenta Mathematicae*, 20:177–190, 1933. doi:10.4064/fm-20-1-177-190.

6 Prosenjit Bose, Jurek Czyzowicz, Evangelos Kranakis, Danny Krizanc, and Anil Maheshwari. Polygon cutting: Revisited. In *Proc. Japanese Conference on Discrete and Computational Geometry (JCDCG)*, volume 1763 of *LNCS*, pages 81–92. Springer, 1998. doi:10.1007/978-3-540-46515-7_7.

7 Hervé Brönnimann and Michael T. Goodrich. Almost optimal set covers in finite VC-dimension. *Discrete and Computational Geometry*, 14(4):463–479, 1995. doi:10.1007/BF02570718.

8 Bernard Chazelle. A theorem on polygon cutting with applications. In *Proc. 23rd IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 339–349, 1982. doi:10.1109/SFCS.1982.58.

9 Hallard T. Croft, Kenneth J. Falconer, and Richard K. Guy. *Unsolved Problems in Geometry*. Springer-Verlag, New York, 1991. doi:10.1007/978-1-4612-0963-8.

10 Gautam Das and Giri Narasimhan. Geometric searching and link distance. In *Proc. 2nd Workshop on Algorithms and Data Structures (WADS)*, volume 519 of *LNCS*, pages 261–272. Springer, 1991. doi:10.1007/BFb0028268.

11 Robert Freimer, Joseph S. B. Mitchell, and Christine Piatko. On the complexity of shattering using arrangements. Technical report, Cornell University, 1991.

12 Unnikrishnan Gopinathan, David J. Brady, and Nikos Pitsianis. Coded apertures for efficient pyroelectric motion tracking. *Opt. Express*, 11(18):2142–2152, 2003. doi:10.1364/OE.11.002142.

13 Roser Guàrdia and Ferran Hurtado. On the equipartition of plane convex bodies and convex polygons. *Journal of Geometry*, 83(1):32–45, 2005. doi:10.1007/s00022-005-0006-0.

14 Steven C. Gustafson. Intensity correlation techniques for passive optical device detection. In *Infrared Technology for Target Detection and Classification*, volume 302, pages 66–70. International Society for Optics and Photonics, SPIE, 1982. doi:10.1117/12.932632.

15 Refael Hassin and Nimrod Megiddo. Approximation algorithms for hitting objects with straight lines. *Discrete Applied Mathematics*, 30(1):29–42, 1991. doi:10.1016/0166-218X(91)90011-K.

16 Tian He, Qiuhua Cao, Liqian Luo, Ting Yan, Lin Gu, John Stankovic, and Tarek Abdelzaher. Electronic tripwires for power-efficient surveillance and target classification. In *Proc. 2nd International Conference on Embedded Networked Sensor Systems (SenSys 2004)*. ACM Press, 2004. doi:10.1145/1031495.1031558.

17 John Hershberger and Subhash Suri. A pedestrian approach to ray shooting: Shoot a ray, take a walk. *J. Algorithms*, 18(3):403–431, 1995. doi:10.1006/jagm.1995.1017.

18 Thomas Jenrich and Andries E. Brouwer. A 64-dimensional counterexample to Borsuk's Conjecture. *Electr. J. Comb.*, 21(4):P4.29, 2014. URL: http://www.combinatorics.org/ojs/index.php/eljc/article/view/v21i4p29.

19 Jeff Kahn and Gil Kalai. A counterexample to Borsuk's conjecture. *Bull. Amer. Math. Soc.*, 29:60–62, 1993. doi:10.1090/S0273-0979-1993-00398-7.

20 Roman Karasev, Alfredo Hubard, and Boris Aronov. Convex equipartitions: the spicy chicken theorem. *Geometriae Dedicata*, 170(1):263–279, 2014. doi:10.1007/s10711-013-9879-5.

21 J. Mark Keil. Polygon decomposition. In Jörg-Rüdiger Sack and Jorge Urrutia, editors, *Handbook of Computational Geometry*, pages 491–518. North Holland/Elsevier, 2000. doi:10.1016/b978-044482537-7/50012-7.

22 Irina Kostitsyna. *Balanced partitioning of polygonal domains*. PhD thesis, Stony Brook University, Stony Brook, NY, 2015.

23 Stefan Langerman and Pat Morin. Covering things with things. *Discrete & Computational Geometry*, 33(4):717–729, 2005. doi:10.1007/s00454-004-1108-4.

24 Anil Maheshwari, Jörg-Rüdiger Sack, and Hristo N. Djidjev. Link distance problems. In Jörg-Rüdiger Sack and Jorge Urrutia, editors, *Handbook of Computational Geometry*, chapter 12, pages 519–558. North-Holland, 2000. doi:10.1016/b978-044482537-7/50013-9.

25 Nimrod Megiddo and Arie Tamir. On the complexity of locating linear facilities in the plane. *Operations Research Letters*, 1(5):194–197, 1982. doi:10.1016/0167-6377(82)90039-6.

26 Joseph S.B. Mitchell, Valentin Polishchuk, and Mikko Sysikaski. Minimum-link paths revisited. *Computational Geometry*, 47(6):651–667, 2014. doi:10.1016/j.comgeo.2013.12.005.

27 R. Nandakumar and N. Ramana Rao. Fair partitions of polygons: An elementary introduction. *Proceedings-Mathematical Sciences*, 122(3):459–467, 2012. doi:10.1007/s12044-012-0076-5.

28 Jack Robertson and William Webb. *Cake-cutting algorithms: Be fair if you can.* AK Peters/CRC Press, 1998.

29 Pablo Soberón. Balanced convex partitions of measures in $\mathbb{R}^d$. *Mathematika*, 58(1):71–76, 2012. doi:10.1112/S0025579311001914.

30 Subhash Suri. On some link distance problems in a simple polygon. *IEEE Trans. Robotics and Automation*, 6(1):108–113, 1990. doi:10.1109/70.88124.

31 Samuel Zahnd, Patrick Lichisteiner, and Tobi Delbruck. Integrated vision sensor for detecting boundary crossings. In *2003 IEEE International Symposium on Circuits and Systems (ISCAS)*, volume 2, 2003. doi:10.1109/ISCAS.2003.1205986.

32 Yunhui Zheng, David J. Brady, and Pankaj K. Agarwal. Localization using boundary sensors: An analysis based on graph theory. *ACM Trans. Sen. Netw.*, 3(4), 2007. doi:10.1145/1281492.1281496.