Finding the Maximum Area Axis-Parallel Rectangle in a Polygon

Karen Daniels * *Victor Milenkovic*† *Dan Roth*‡
Harvard University Division of Applied Sciences,
Center for Research in Computing Technology,
Cambridge, MA 02138.

1 Introduction

We consider the geometric optimization problem of finding the maximum area axis-parallel rectangle (MAAPR) in an n-vertex general polygon[1]. We characterize the MAAPR in a general polygon and give an $O(n\alpha(n)\log^2 n)$ time divide-and-conquer algorithm for finding it, where $\alpha(n)$ is the slowly growing inverse of Ackermann's function. This is based on algorithms we present for two other polygon classes: horizontally (vertically) convex polygons, for which we give an $O(n\alpha(n)\log n)$ time divide-and-conquer algorithm, and orthogonally convex polygons[2], for which we present an $O(n\alpha(n))$ time algorithm. We also give a $\Theta(n)$ time algorithm for xy-monotone polygons[3].

We prove a lower bound of time in $\Omega(n \log n)$ for two classes of polygons: self-intersecting polygons, and polygons with holes. These contrast with the $\Theta(n)$ results achievable for the corresponding enclosure problems.

Our rectangle problem arises naturally in applications where an inexpensive internal approximation to a polygon is desired. It is useful, for example, in the industrial problem of laying out apparel pattern pieces on clothing "markers" with minimal cloth waste [9, 10]. Despite its practical importance, work on finding the MAAPR has been restricted to rectilinear polygons [3, 8, 15] and, recently, convex polygons [4]. Wood and Yap [15] note that the MAAPR in a simple rectilinear polygon can be found using an algorithm for the largest empty rectangle problem,[4] which can be solved in $O(n\log^2 n)$ time [2]. For a constrained type of rectilinear polygon, Aggarwal [3] gives a $\Theta(n)$ time algorithm for finding the MAAPR using the total monotonicity of the area matrix associated with the polygon. Amenta [4] has shown that the MAAPR in a convex polygon can be found in linear time by phrasing it as a convex programming problem. No algorithm is known for finding the MAAPR in a general polygon, nor has a lower bound tighter than $\Omega(n)$ been established. We present the first results for the general case.

Our paper is organized as follows. In Section 2 we characterize the MAAPR for general polygons by considering different cases based on the number of corners of the rectangle contacting the boundary of the polygon. Based on this characterization we show, in Section 3, how to reduce key subcases of the MAAPR problem to finding the maximal element in an area matrix corresponding to a pair of diagonally opposite xy-monotone chains. Different visibility conditions imposed by the polygon on the chains lead to different types of monotone matrices. In Section 4 we give efficient solutions to all of the associated matrix problems by building on the work of Klawe and Kleitman [7], and Aggarwal [3]. This leads to the algorithmic results of Section 5: $\Theta(n)$ time for xy-monotone polygons based on properties of totally monotone area matrices, $O(n\alpha(n))$ time for orthogonally convex polygons based on completion techniques for monotone area matrices, $O(n\alpha(n)\log n)$ time for horizontally (vertically) convex polygons, and $O(n\alpha(n)\log^2 n)$ time for the general case. The general algorithm contains two levels of divide-and-conquer: the highest level deals with horizontally convex polygons and the lowest with orthogonally convex polygons.

In Section 6 we prove a lower bound of time in $\Omega(n \log n)$ for finding the MAAPR in a self-intersecting polygon, in contrast to the $\Theta(n)$ result achievable for the corresponding enclosure problem. This establishes a separation in the running time of the "dual problems" and demonstrates a limit to their duality. We also give a lower bound of time in $\Omega(n \log n)$ for finding the MAAPR in a polygon with holes; this yields both an

*This research was funded by the Textile/Clothing Technology Corporation from funds awarded to them by the Alfred P. Sloan Foundation.

†This research was funded by the Textile/Clothing Technology Corporation from funds awarded to them by the Alfred P. Sloan Foundation and by NSF grants CCR-91-157993 and CCR-90-09272.

‡Supported by NSF grants CCR-89-02500 and CCR-92-00884 and by DARPA AFOSR-F4962-92-J-0466.

[1] A general polygon is a polygonal region in the plane with n vertices and an arbitrary number of components and holes.

[2] An orthogonally convex polygon is both horizontally and vertically convex. This class contains the class of convex polygons.

[3] A simple polygon consisting of two xy-monotone chains is an xy-monotone polygon.

[4] Given a rectangle containing n points, find the largest area subrectangle with sides parallel to those of the original rectangle which contains none of the given points ([6, 11, 2]).

upper and lower bound for such polygons. Both proofs involve a reduction from MAX-GAP.

2 Characterizing the MAAPR

In this section we characterize the MAAPR R contained in a general polygon P by considering different cases based on the number of corners of R on the boundary of P, and outline a naive algorithm for finding the MAAPR based on this characterization. Intuitively, if R is inside P, it can "grow" until each of its four sides is stopped by contact with the boundary of P. In order to discuss contacts between R and P, we require the notion of a *reflex extreme* vertex, introduced in [14].

Definition 2.1 *A reflex vertex v of P is a reflex extreme if there exists a vertical or horizontal line of support of v that is interior to P.*

Using this definition, we can characterize contacts between R and P as being of two types: 1) an edge of R with a vertex of P, and 2) a vertex of R with an edge of P. In the first case, a reflex extreme vertex of P touches an edge of R and stops growth in one direction; we call this a *reflex contact*. Two reflex contacts with adjacent sides of R are *adjacent* reflex contacts which fix a corner of R. In the second case, a corner of R touches an edge of P, forming a *sliding contact*.

By enumerating the sliding contacts between R and P we derive the set of six cases given in Table 1.

Case	# corners of R on P (sliding contacts)	min # reflex extreme vertices of P touching R (reflex contacts)
(0)	0	4
(1)	1	2
(2a)	2 opposite	0
(2b)	2 adjacent	1
(3)	3	0
(4)	4	0

Table 1: Characterization of MAAPR

Theorem 2.1 *The maximum area axis-parallel rectangle R of a general polygon P conforms to one of the six cases given in Table 1.*

Proof: R has either 0, 1, 2, 3, or 4 corners on the boundary of P. For each possibility, we show that the minimal number of reflex contacts determining the MAAPR is given by the associated case(s) in Table 1. We base our argument on the observation that R is maximal if no horizontal or vertical growth is possible. It is sufficient to fix two opposite corners of R using a pair of adjacent reflex contacts. It is also sufficient to fix only one corner of R when the opposite corner has a sliding contact with an edge of P. In this case R can be found by parameterizing the edge and maximizing a quadratic in one variable; we denote this the *1-parameter problem*. Further relaxation is also possible: the problem of finding R when both opposite corners have sliding contacts with edges of P is a *2-parameter problem*. In the full version we prove the following claim which allows us to reduce a 2-parameter problem to four 1-parameter problems. Thus, R is uniquely determined in both 1 and 2-parameter problems.

Claim 2.2 *Let $e_1 = (u_1, v_1), e_2 = (u_2, v_2)$ be two edges of P, and assume that R is the MAAPR with opposite corners on e_1, e_2, and no other contacts with P. Then, at least one of these corners coincides with an endpoint of an edge, i.e. is on one of u_1, u_2, v_1, v_2.*

Now we analyze the different cases corresponding to the number of sliding contacts. Case 0 has no sliding contacts, so each corner of R must be fixed using reflex contacts. This requires four reflex contacts. Case 1 has one sliding contact. Adding only one reflex contact allows R to grow in one direction, so two are necessary. Two are sufficient to determine R, since this is a 1-parameter problem. In case 2, the two sliding contacts can either be opposite or adjacent. The former is case 2a, which is a 2-parameter problem, so the two sliding contacts determine R. In the latter case, 2b, two adjacent corners of R with sliding contacts on edges of P share an edge e of R. We must add a reflex contact with the edge opposite e to determine R. It is easily shown that this reduces to a 1-parameter problem. Case 3 is a constrained version of case 2a. The extra constraints induced by the adjacent pairs of sliding contacts reduce this to a 1-parameter problem. No reflex contacts are required. In case 4, a set of four sliding contacts associated with corners of R yields a system of four equations in four unknowns, which determines a unique axis-parallel rectangle, if one exists. ■

Based on this characterization one can find the MAAPR contained in a polygon by finding the MAAPR under the constraints of each of the six cases and selecting the largest one.

Theorem 2.3 *For a given set of vertex/edge contacts between R and P from Table 1, the MAAPR can be found in constant time.*

Proof: In case 0, all four corners of the MAAPR are uniquely determined by the four reflex extreme vertices. In case 4, the system of four equations in four unknowns can be solved in constant time. The remaining cases rely on the 1-parameter problem, which can be solved in constant time because it only involves maximizing a quadratic in one variable. By the proof of Theorem 2.1, case 1 is a 1-parameter problem, and cases 2b and 3 reduce to a 1-parameter problem. For case

2, by Claim 2.2, a 2-parameter problem can be solved by comparing the results of four 1-parameter problems, and so can also be solved in constant time. Thus, all six cases can be solved in constant time. ∎

An interesting alternate constant-time solution to the 1-parameter problem is based on the following lemma, which we prove in the full paper.

Lemma 2.4 *Given a point p and a line L with slope s, the MAAPR with one corner at p and opposite corner at point q on L has diagonal* $\overline{pq}$*, where the slope of* $\overline{pq} = -s$.

A naive MAAPR algorithm can use Theorem 2.3 and supply it, in each case, all possible sets of candidate edges of P. With $O(n)$ preprocessing time to build horizontal and vertical visibility maps [5], we can determine in $O(1)$ time, given a set of edges for a case, the pieces of those edges that are candidates for determining the MAAPR. We conclude:

Theorem 2.5 *The MAAPR of an n-vertex general polygon can be found in* $O(n^4)$ *time.*

In the remainder of the paper we show how to use the MAAPR characterization combined with fast matrix searching to develop a more sophisticated approach to this problem.

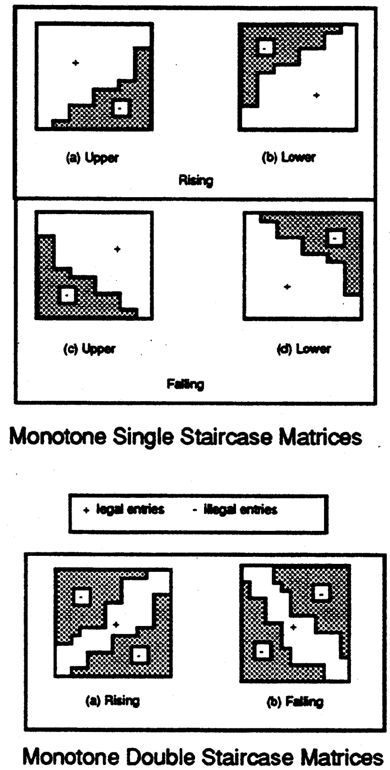

Figure 1: Types of Staircase Matrices

3 MAAPR as a Matrix Problem

In this section we explore the relationship between finding the MAAPR and the theory of monotone matrices. We show that, for diagonally opposite xy-monotone chains with certain visibility restrictions, the problem of computing the MAAPR can be reduced to finding the maximal element in partially monotone area matrices we construct for the chains. We define below the subclasses of matrices we consider (see also Figure 1):

Definition 3.1 *In a real* $n \times m$ *matrix* M*, let* $k(i)$ *be the index of the leftmost column containing the maximum value in row i. M is* **monotone**[1] *if* $i_1 > i_2$ *implies that* $k(i_1) \geq k(i_2)$*. M is* **totally monotone** *if all its submatrices are monotone. (It is sufficient for all* 2×2 *submatrices to be monotone.)*

Definition 3.2 *A matrix M is* **monotone-single-staircase**[2] *if there exists one special set of entries S such that any* 2×2 *minor that does not contain entries from S is totally monotone; the boundary of S forms an xy-monotone staircase inside M.*

In Figure 1, the special set of entries is shaded; we call them *illegal* entries. The figure distinguishes between rising and falling staircase matrices. In addition, *upper* and *lower* refer to the position of the illegal entries.

Definition 3.3 *A matrix M is* **monotone-double-staircase**[2] *if there exist two special sets of entries* S_1 *and* S_2 *such that any* 2×2 *minor that does not contain entries from those sets is totally monotone; the boundary of each set forms a staircase inside M and the two sets of entries lie in diagonally opposite corners of M.*

Given two polygonal chains C_1 and C_2 that are non-intersecting and have the same slope direction, we associate with them two *area matrices* M_{C_1,C_2} and M_{C_2,C_1} as follows: in M_{C_1,C_2} entry m_{ij} corresponds to the area of the MAAPR which has opposite corners on the i-th vertex of C_1 and on the j-th edge of C_2. Similarly, in M_{C_2,C_1} entry m_{ij} corresponds to the area of the MAAPR which has opposite corners on the i-th vertex of C_2 and the j-th edge of C_1. We calculate areas as follows: if C_1 and C_2 both have negative slope then, for $p \in C_1$ and $q \in C_2$, the area of the associated rectangle is $(p_x - q_x)(p_y - q_y)$. If both chains have positive slope, the area has opposite sign. Note that this forces the area to be *positive* if and only if p and q are *rectangularly visible*[5]. In the following theorems we reduce the computation of the MAAPR to a matrix searching problem.

[5]Two points p and q have *rectangular visibility* [12] if the interior of the axis-parallel rectangle with diagonal $\overline{pq}$ does not intersect C_1 or C_2.

Theorem 3.1 *Let C_1 and C_2 be two weakly L_1-visible polygonal chains. Then:*
(i) The area of the MAAPR which has opposite corners on C_1 and C_2 is the maximum of the maximal elements in M_{C_1,C_2} and M_{C_2,C_1}.
(ii) M_{C_1,C_2} and M_{C_2,C_1} are both totally monotone matrices.

Proof: (Sketch) The proof of (i) is based on the MAAPR characterization (see Claim 2.2). When the chains C_1 and C_2 are rectilinear the corners of all candidate rectangles are among the original set of vertices, and the claim is trivially true. In the nonrectilinear case, this does not hold. However, finding the area of the MAAPR with opposite corners on edge e_i of C_1 and edge e_j of C_2 is a 2-parameter problem which reduces to four 1-parameter problems.

The proof of (ii) consists of two steps. For both steps assume, w.l.o.g., both chains have negative slope. W.l.o.g. we prove (ii) for M_{C_1,C_2}, so consider two vertices v_i and $v_{i'}$ on C_1 and two edges e_j and $e_{j'}$ on C_2. The MAAPR for v_i with e_j has a vertex v_{ij} on e_j; call v_{ij} the MAAPR vertex for v_i. Similarly, let the MAAPR vertex of v_i with $e_{j'}$ be $v_{ij'}$, the MAAPR vertex of $v_{i'}$ with e_j be $v_{i'j}$, and the MAAPR vertex of $v_{i'}$ with $e_{j'}$ be $v_{i'j'}$. Denote the area of the rectangle with opposite corners on two vertices p and q by $A(p,q)$. In the first step, we use Lemma 2.4 to show that the vertices on C_2 have the order: v_{ij}, $v_{i'j}$, $v_{ij'}$, and $v_{i'j'}$. In the second step, we show M_{C_1,C_2} is totally monotone by proving that the 2×2 submatrix associated with vertices v_i, $v_{i'}$ and edges e_j, $e_{j'}$ is totally monotone. It suffices to show that: $A(v_i,v_{ij'}) > A(v_i,v_{ij}) \Rightarrow A(v_{i'},v_{i'j'}) > A(v_{i'},v_{i'j})$. To show this we need the intermediate result: $A(v_i,v_{ij'}) > A(v_i,v_{i'j}) \Rightarrow A(v_{i'},v_{ij'}) > A(v_{i'},v_{i'j})$, which is proven using the relationships among the x and y coordinates obtained from the ordering of vertices along the two chains. We then take advantage of the MAAPRs to show that: $A(v_i,v_{ij}) \geq A(v_i,v_{i'j})$ and $A(v_{i'},v_{i'j'}) \geq A(v_{i'},v_{ij'})$. Combining these two MAAPR inequalities with the intermediate result and using transitivity establishes (ii). ∎

Consider now two diagonally opposite xy-monotone chains C_1 and C_2 of an orthogonally convex polygon. We must take into account the other pair of chains, because they can block visibility and ruin the total monotonicity of the area matrices by forcing entries to become illegal. For this case we show:

Theorem 3.2 *Let C_1 and C_2 be two diagonally opposite xy-monotone chains of an orthogonally convex polygon. Then:*
(i) The area of the MAAPR which has opposite corners on C_1 and C_2 is the maximum of the maximal elements in M_{C_1,C_2} and M_{C_2,C_1}.
(ii) M_{C_1,C_2} and M_{C_2,C_1} are both monotone rising double staircase matrices.

Proof: (Sketch) Let b,t,l,r denote the highest, lowest, leftmost, and rightmost points on P, respectively. We consider w.l.o.g. the opposite chains lt and br. The proof of (i) is again based on Claim 2.2. To establish (ii), form the rectangle R_1 with t and r as opposite corners. Projecting points from lt horizontally onto R_1, and points from br vertically onto R_1, defines a rectangular grid on R_1. By following tr across and down the grid we can determine how chain tr restricts the rectangular visibility of chain lt with chain br. This yields a monotone upper rising single staircase matrix corresponding to the effect of tr. An analogous monotone lower rising single staircase matrix can be constructed corresponding to the effect of lb. It is easily shown that the intersection[6] of the legal entries of a monotone upper rising single staircase matrix and a monotone lower rising single staircase matrix is a monotone rising double staircase matrix. ∎

4 Solving the Matrix Problem

In this section we show how to efficiently compute the maximal elements in various classes of monotone and partially monotone matrices. The main result of this section is Lemma 4.3, which involves a monotone rising double staircase matrix. This result forms the basis of the algorithms presented in Section 5. Lemma 4.3 relies on the other results cited below, and on matrix completion techniques. Completion returns a legal value instead of an illegal one during a matrix element query without affecting the maximum value in the matrix. That is, given a non-totally monotone matrix in interval form (i.e. as a set of n intervals, each corresponding to the connected set of legal entries in a row), we can operate on it as if it were totally monotone. Completion via increasing and decreasing sequences appears in [7]. Here we introduce completion by propagation, which involves propagating the values at the endpoints of a legal interval to the left and right. Propagation does not introduce new row minima or maxima, an advantage it has over completion via increasing or decreasing sequences. Details on completing different types of matrices appear in the full paper.

Lemma 4.1 ([1]) *If any entry of a totally monotone matrix of size $m \times n$ can be computed in constant time, then the row-maximum problem for this matrix can be solved in $\Theta(m+n)$ time.*

[6] A matrix M is the intersection of M_1 and M_2 if the set of legal entries of M is the intersection of the legal entries of M_1 and M_2.

Lemma 4.2 ([7]) *If any entry of a monotone single staircase matrix of size* $n \times m$ *can be computed in constant time, then the row-maximum problem for this matrix can be solved in* $O(n\alpha(m) + m)$ *time.*

Lemma 4.3 *Given a monotone rising double staircase matrix in interval form, the row-maximum problem for this matrix can be solved in* $O(n\alpha(m) + m)$ *time.*

Proof: (Sketch) The matrix is the intersection of the legal entries of a monotone upper rising single staircase matrix and a monotone lower rising single staircase matrix. Completing the upper left portion of the matrix via propagation transforms it into a monotone upper rising single staircase matrix, whose maximum element can be found in $O(n\alpha(m) + m)$ time via Lemma 4.2. ■

5 Finding the MAAPR

In this section we use the results of the previous sections to derive efficient algorithms for computing the MAAPR in different classes of n-vertex polygons. We use the characterization results of Section 2 to reduce the MAAPR problem to the corresponding matrix problems discussed in Section 3 and then use the results of Section 4 to give the computational bounds. We first present our results for subclasses of polygons, leading to the general result in Theorem 5.4. For general polygons we use two levels of divide-and-conquer, which require finding the MAAPR in an orthogonally convex polygon at the lowest level. This follows the divide-and-conquer approach used by McKenna and O'Rourke [8] to find the largest axis parallel rectangle in a rectilinear polygon. They obtain a rectilinear orthogonally convex polygon at the lowest level. However, they find the largest rectangle in this polygon in $O(n \log^3 n)$ time, whereas we solve the problem for orthogonally convex polygons in $O(n\alpha(n) \log n)$ time.

Theorem 5.1 *The MAAPR in an xy-monotone polygon can be found in time in* $\Theta(n)$.

The proof follows immediately from Theorem 3.1 and Lemma 4.1.

Theorem 5.2 *The MAAPR in an orthogonally convex polygon can be found in* $O(n\alpha(n))$ *time.*

Proof: (Sketch) In the first part of the proof we show that the MAAPR in an orthogonally convex polygon P is either of type 2a, 3, or 4. We then show that case 2a can be solved in $O(n\alpha(n))$ time and cases 3 and 4 can be solved in linear time.

In case 2a, the boundary of P can be partitioned, in linear time, into four xy-monotone polygonal chains. Now the problem can be reduced to an application of Theorem 3.2 and Lemma 4.3. Note that, in the proof of Theorem 3.2, a projected xy-monotone chain cuts through the rectangular grid, intersecting a linear number of rectangles. By following the chain across (and down) the grid, we can, in linear time, determine how it restricts the rectangular visibility. For each pair of diagonally opposite chains two monotone rising double staircase matrices are produced, which together form a monotone rising double staircase matrix. Therefore, we can solve case 2a by Lemma 4.3 in time in $O(n\alpha(n))$.

In cases 3 and 4, the proof rests on the fact that we can, in linear time, obtain horizontal and vertical visibility maps, then check for 3 or 4 vertices of R on P in linear time using a sweep algorithm. Two sweeps in y (top-down and bottom-up) suffice to solve cases 3 and 4, since, in both cases, either the top or bottom corners of R are on edges of P. Since the four chains are naturally ordered in x and y, no sorting is required. ■

Theorem 5.3 *The MAAPR in a horizontally (or vertically) convex polygon can be found in* $O(n\alpha(n) \log n)$ *time.*

Proof: (Sketch) W.l.o.g. we treat the horizontally convex case. We use a divide-and-conquer algorithm which partitions the vertices of the polygon P using a horizontal line L into two sets of roughly equal size. Suppose L is partitioned into k pieces $L_1, L_2, \ldots, L_k$ by the polygon. The merge step requires that we find the MAAPR intersecting $L_i, 1 \leq i \leq k$. For a given L_i, we denote this R_L. We denote the largest polygon containing L_i that is monotone with respect to L_i by V. V is orthogonally convex. We prove in the full paper that $R_L \subseteq V$, that V has $O(n)$ vertices, and it can be constructed in $\Theta(n)$ time. Now we find the MAAPR in V, which, by Theorem 5.2 can be done in $O(n\alpha(n))$ time. This yields a recurrence of the form $T_1(n) = T_1(n/2) + O(n\alpha(n))$, which gives an $O(n\alpha(n) \log n)$ algorithm for finding the MAAPR of P. ■

Theorem 5.4 *The MAAPR in a general polygon can be found in* $O(n\alpha(n) \log^2 n)$ *time.*

Proof: (Sketch) The algorithm uses two phases of divide-and-conquer. The first partitions with a vertical line. Its merge step requires finding the MAAPR in a horizontally convex polygon, which, by Theorem 5.3 can be done in $O(n\alpha(n) \log n)$ time. We have the recurrence: $T_2(n) = T_2(n/2) + O(n\alpha(n) \log n)$, which gives $O(n\alpha(n) \log^2 n)$. ■

6 Lower Bounds

We prove a lower bound of time in $\Omega(n \log n)$ for finding the MAAPR in a self-intersecting polygon, and for polygons with holes[7]. This contrasts with the $\Theta(n)$ result achievable for the corresponding enclosure problem, thus establishing a separation in the running time of these two dual problems and demonstrating a limit to their duality[8].

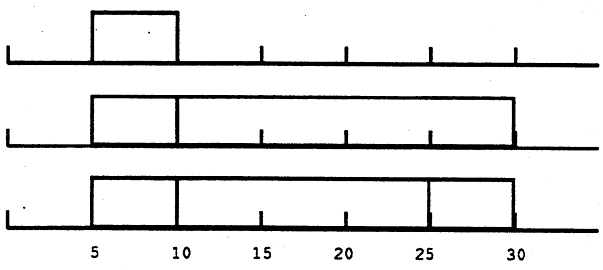

Figure 2: Rectilinear polygon constructed for input = 10, 5, 30, 25

Theorem 6.1 *Finding the MAAPR in an n-vertex self-intersecting polygon requires time in $\Omega(n \log n)$ in both the linear and algebraic decision tree models.*

Proof: (Sketch) We reduce the MAX-GAP problem[9] [3] to the MAAPR problem for self-intersecting rectilinear polygons. Consider an instance of MAX-GAP: given a set of n real numbers $x_1, x_2, ..., x_n$, we must find the maximum difference between two consecutive numbers in the sorted list . We construct from this set, in linear time, a self-intersecting rectilinear polygon of unit height as follows: each x_i in the sequence corresponds to a rectangle $r_i = [(x_1, 0), (x_1, 1), (x_i, 1), (x_i, 0)]$. We start the construction from $(x_1, 0)$, and complete the degenerate rectangle r_1, then constructing $r_2, \ldots, r_n$ (as shown in Figure 2). This construction results in a self-intersecting polygon, with the property that the area of the MAAPR included in it is the solution to the corresponding MAX-GAP problem, thus proving the theorem. ∎

Theorem 6.2 *Finding the MAAPR in an n-vertex polygon with holes requires time in $\Omega(n \log n)$ in both the linear and algebraic decision tree models.*

The proof involves a reduction from MAX-GAP, and is omitted.

[7] This result was suggested by Binhai Zhu, and is similar to the proof provided by McKenna and O'Rourke [8] that finding the largest axis-parallel rectangle in a rectilinear polygon *with holes* requires $\Omega(n \log n)$ time.

[8] It is interesting to note that the dual problems of largest empty circle and smallest enclosing circle for a set of points also have different lower bounds. The largest empty circle can be constructed in $\Theta(n \log n)$ time, and the smallest enclosing circle can be found in $\Theta(n)$ time [13].

[9] In both the linear and algebraic decision tree models (if not enhanced to include floor and ceiling functions), MAX-GAP has a lower bound of $\Omega(n \log n)$.

Acknowledgments

The authors gratefully acknowledge the work of Binhai Zhu on the lower bound result for self-intersecting polygons. We are also grateful for the helpful comments made by Alok Aggarwal, Pankaj Agarwal, Zhenyu Li and Marios Mavronicolas, and background information provided by David Dobkin and Joseph O'Rourke.

References

[1] A. Aggarwal, M.M. Klawe, S. Moran, P. Shor, and R. Wilber. Geometric applications of a matrix-searching algorithm. *Algorithmica*, 2:195–208, 1987.

[2] A. Aggarwal and S. Suri. Fast Algorithms for Computing the Largest Empty Rectangle. In *Proceedings of the 3rd ACM Symposium on Computational Geometry*, pages 278 – 290, 1987.

[3] A. Aggarwal and J. Wein. *Computational Geometry Lecture Notes for MIT 18.409*. 1988.

[4] N. Amenta. Largest Volume Box is Convex Programming. *private communication*, 1992.

[5] B. Chazelle. Triangulating a Simple Polygon in Linear Time. In *Proceedings of the Thirty-First Annual Symposium on Foundations of Computer Science*, pages 220–230, 1990.

[6] B. Chazelle, R.L. Drysdale III, and D.T. Lee. Computing the largest empty rectangle. *SIAM J. Comput.*, 15:300–315, 1986.

[7] M. M. Klawe and D. J. Kleitman. An Almost Linear Time Algorithm for Generalized Matrix Searching. *SIAM Journal of Discrete Mathematics*, 3(1):81–97, 1990.

[8] M. McKenna, J. O'Rourke, and S. Suri. Finding the largest rectangle in an orthogonal polygon. In *Proceedings of the 23rd Allerton Conference on Communication, Control, and Computing*, pages 486–495, 1985.

[9] V. Milenkovic, K. Daniels, and Z. Li. Automatic Marker Making. In *Proceedings of the Third Canadian Conference on Computational Geometry*, 1991.

[10] V. Milenkovic, K. Daniels, and Z. Li. Placement and Compaction of Nonconvex Polygons for Clothing Manufacture. In *Proceedings of the Fourth Canadian Conference on Computational Geometry*, 1992.

[11] A. Naamad, W.L. Hsu, and D.T. Lee. On maximum empty rectangle problem. *Discrete Appl. Math.*, 8:267–277, 1984.

[12] M.H. Overmars and D. Wood. On rectangular visibility. *J. Algorithms*, 9:372–390, 1988.

[13] F. Preparata and M. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, 1985.

[14] S. Schuierer, G.J. E. Rawlins, and D. Wood. A generalization of staircase visibility. In *Proceedings of the 3rd Canadian Conf. Comput. Geom.*, pages 96–99, 1991.

[15] D. Wood and C. K. Yap. The orthogonal convex skull problem. *Discrete Computational Geometry*, 3:349–365, 1988.