Finding the Largest Axis-Aligned Rectangle in a Polygon in $O(n \log n)$ time

Ralph P. Boland*† Jorge Urrutia‡

Abstract

We consider the problem of finding the largest area axis-aligned rectangle contained in an n vertex polygon. We present an algorithm that solves this problem in $O(n \log n)$ time. This is an improvement by a factor of $O(\log n)$ over the best known algorithm. Our method of achieving this improvement is noteworthy. The previous algorithm constructs a cover of the polygon by a collection of subpolygons, of the type we call SAM polygons, such that any rectangle in the polygon is contained in one of the SAM subpolygons. This cover requires $O(n \log n)$ time and space to construct. Our algorithm also constructs a cover of the polygon by SAM subpolygons such that any rectangle in the polygon is contained in one of the subpolygons but does so using only linear time and space.

1 Introduction

An axis-aligned rectangle is a rectangle whose sides are parallel to the x-axis or y-axis. In this paper we present an algorithm for finding the largest axis-aligned rectangle contained in a simple polygon. Henceforth, all rectangles used in this paper are assumed to be axis-aligned. As in [4], we refer to this problem as the *LR problem*. In [4], the authors showed how to solve the LR problem on an n vertex simple polygon P in $O(n \log^2 n)$ time. Their method generates a set of simpler subpolygons of P such that any rectangle in P is contained in one of them. It then finds the largest rectangle in each of the simpler subpolygons and selects the largest one. In this paper we solve the LR problem using the same approach except that the total size of the simpler subpolygons that we generate is smaller by a factor of $O(\log n)$; the result is an algorithm that solves the LR problem in $O(n \log n)$ time.

For any set of polygons $\wp$ we denote $\|\wp\| = \sum_{Q_i \in \wp} |Q_i|$. Let Π be a cover of P by subpolygons. Let R be any rectangle contained in P such that the number of elements of Π needed to cover R is maximized. We denote this number by $\mathtt{C}_{\square}(\Pi)$. Using the terminology in [1], we call $\mathtt{C}_{\square}(\Pi)$ the *rectilinear visibility cover number* of Π. If $\mathtt{C}_{\square}(\Pi) = o(|\Pi|)$ then we call Π a *rectilinear visibility cover* of P. All covers in this paper, that are not partitions, will have a rectilinear visibility cover number of 1. Both this paper and [4] use the following observation.

Observation 1.1. *Assume, for some function g, that the LR problem can be solved, for any element Q of Π, in $O(|Q| * g(|Q|))$ time. If $\mathtt{C}_{\square}(\Pi) = 1$ and Π can be constructed in $O(\|\Pi\| * g(n))$ time then the LR problem can be solved on P in $O(\|\Pi\| * g(n))$ time.*

Proof. First, build Π in $O(\|\Pi\| * g(n))$ time. Then find a largest rectangle in each element of Π in $\sum_{Q_i \in \Pi} O(|Q_i| * g(|Q_i|)) = O(\|\Pi\| * g(n))$ time and select the largest one. □

We call a polygon that is either a y-monotone polygon that admits a vertical chord joining a highest point on its boundary with a lowest point on its boundary or an x-monotone polygon that admits a horizontal chord joining a leftmost point on its boundary with a rightmost point on its boundary a *separated axis monotone polygon* or *SAM polygon*.[1] The following lemma is proved in [4].

Lemma 1.2. *The LR problem for any m vertex SAM polygon can be solved in $O(m \log m)$ time.*

In [4], the authors first showed how to construct a cover, say Π, of P, such that $\|\Pi\| = O(n \log n)$, the elements of Π are SAM polygons, and $\mathtt{C}_{\square}(\Pi) = 1$. Thus, by Observation 1.1 and Lemma 1.2, the LR problem on P can be solved in $O(n \log^2 n)$ time. Their algorithm also works for polygons with holes and in the same time complexity (where n is then the total number of vertices in P and its holes). They also provided a survey of previous results on this problem and gave some applications for the case that there are holes in P.

Our contribution to this problem is a method of constructing a cover, say Π', of P such that $\mathtt{C}_{\square}(\Pi') = 1$ and the elements of Π' are SAM polygons and, if P is not rectilinear, quadrilaterals; note that quadrilaterals cause no negative consequences since the LR problem is easily solved on a quadrilateral. Our improvements,

*University of New Brunswick, Canada
email: rboland@unb.ca

†Funded in part by NSERC and the University of New Brunswick

‡Instituto de Mathematicas Universidad Nacional Autonoma de Mexico
email: urrutia@math.unam.mx

[1] In [4] these polygon classes are called vertically separated horizontally convex polygons and horizontally separated vertically convex polygons respectively.

over the algorithm in [4], are that $\|\Pi'\| = O(n)$ and our algorithm constructs Π' in linear time. Thus, by Observation 1.1 and Lemma 1.2, the LR problem on P can be solved, using our algorithm, in $O(n \log n)$ time; an improvement by a factor of $O(\log n)$.

In order to construct Π' it is necessary to construct a horizontal trapezoidization of P and a vertical trapezoidization of P. Either trapezoidization of P can be constructed in linear time by a complex algorithm [2] or in $O(n \log n)$ time by a number of algorithms; see for example [3], [5], and [6]. If an $O(n \log n)$ time trapezoidization algorithm is used then the overall time complexity of our algorithm remains at $O(n \log n)$.

All polygons used in this paper are assumed to be simple and without adjacent collinear edges. We need some basic terminology. Let $P = \{v_0, v_1, ..., v_{n-1}\}$ be any polygon. We assume that P has a counterclockwise orientation and that the edges of P are directed accordingly. To simplify matters, P is assumed to have an edge, denoted base(P) and called the *base edge* of P and furthermore, base(P) $= \overline{v_0 v_1}$. Unless otherwise stated base(P) is assumed to be axis aligned. If necessary, base(P) may have length 0.

Let γ be any chord of P. We denote by P_γ the subpolygon of P determined by γ that does not contain v_0; if v_0 is an endpoint of γ then P_γ denotes the subpolygon that contains v_{n-1} on its boundary. Let ϕ be a chord or edge of P. We denote by $\text{vis}_\perp(\phi, P)$ the subpolygon, say Q, of P such that, for any point p interior to Q, there is a point $q \in \phi$ where the line segment $\overline{pq}$ is perpendicular to ϕ and interior to P. When P is implied we abbreviate this to $\text{vis}_\perp(\phi)$. The following idea is used in [4] and by our algorithm.

Observation 1.3. *Let γ be any axis-aligned chord of a polygon. Then $\text{vis}_\perp(\gamma)$ is a SAM polygon.*

The remainder of this paper is organized as follows.

In Section 2 we define rectilinear tree partitions. We then show, for any rectilinear tree partition, say Π, of a polygon P, that any rectangle interior to P that intersects more than one element of Π must be contained in one of a set of SAM subpolygons of P determined by the chords used to construct Π. This set of SAM subpolygons can be constructed in linear time and space.

In Section 3 we solve the LR problem for rectilinear polygons using a kind of rectilinear tree partition called a histogram partition; see [8].

In Section 4 we extend the results of the previous section to general polygons.

In Section 5 we state our final results. We then discuss future research directions.

2 Rectilinear Tree Partitions

Let P be a polygon and Q be any subpolygon of P such that the edges of Q are either chords of P or parts of edges of P; however, either all or none of the interior of base(P) is on the boundary of Q. Then we call Q a *cut subpolygon* of P. We further denote by door(Q) the chord of P on the boundary of Q such that Q is contained in $P_{\text{door}(Q)}$; if there is no such chord then door(Q) = base(P). We denote by windows(Q) the remaining edges of Q that are chords of P and refer to them as the *windows* of Q. We assume that the endpoints of door(Q) are the first two vertices of Q visited during a counterclockwise traversal of Q and that base(Q) = door(Q).

Let U be any partition of P into cut subpolygons. We call U a *tree partition* of P. This term is appropriate since the graph in which each vertex corresponds to an element of U and the edges of the graph correspond to coincident elements of U is a tree. We denote by $\widehat{\Gamma}(U)$ the chords of P used to construct U. If the elements of $\widehat{\Gamma}(U)$ are all axis-aligned, and no axis-aligned chord of P intersects both the door and a window of the same element of U then we call U a *rectilinear tree partition*. Rectilinear tree partitions have a number of properties useful in solving the LR problem. Let Π be any rectilinear tree partition of P.

Lemma 2.1. *Any axis-aligned chord of P intersects at most 2 elements of $\widehat{\Gamma}(\Pi)$ and 3 elements of Π.*

Proof. Let γ be any axis-aligned chord of P such that γ intersects at least 2 elements of Π and let T be any element of Π such that γ intersects a window of T. Note that, for any element Q of Π, γ cannot intersect both door(Q) and any element of windows(Q), nor can γ intersect more than 2 windows of Q. Therefore, since the windows of T are doors of other elements of Π, it must be that γ cannot intersect any window of any element of Π except those of T. Thus, γ can intersect at most 2 elements of $\widehat{\Gamma}(\Pi)$ and so γ can intersect at most 3 elements of Π. □

Let $S = \{\text{vis}_\perp(\gamma) : \gamma \in \widehat{\Gamma}(\Pi)\}$ with duplicates allowed and $S' \subset S$ be the set of elements of S that are interior to other elements of S; in the case that S contains 2 copies of the same subpolygon only one is put into S'. We denote SAMPC$(\Pi) = S - S'$; see Fig. 1.

Lemma 2.2. $\complement_\square(\mathit{SAMPC}(\Pi) \cup \Pi) = 1$.

Proof. Let R be any rectangle contained in P but not contained in any element of Π. Then R intersects some chord $\gamma \in \widehat{\Gamma}(\Pi)$. But then $R \subseteq \text{vis}_\perp(\gamma)$ since 2 of the edges of R are perpendicular to γ. But $\text{vis}_\perp(\gamma) \in$ SAMPC(Π) so R is covered by a single subpolygon of SAMPC$(\Pi) \cup \Pi$. □

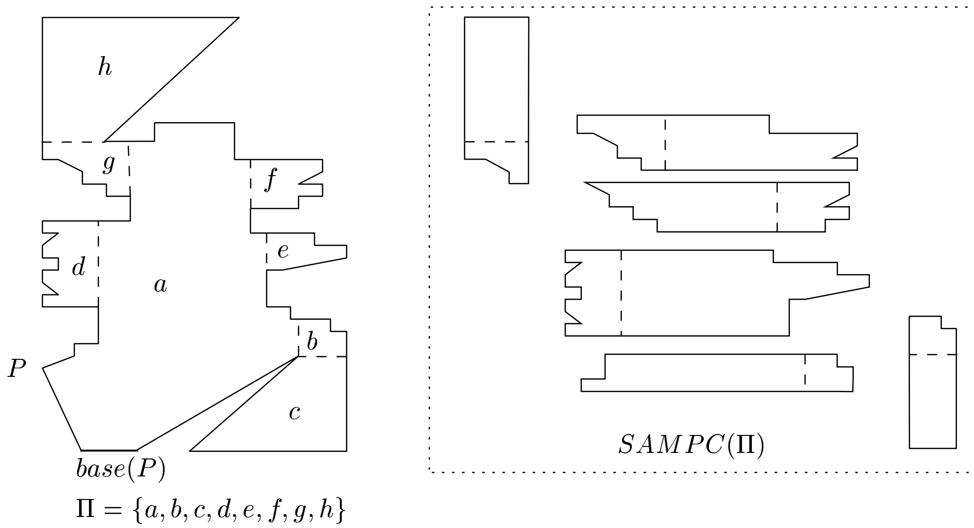

Figure 1: *A rectilinear tree partition* Π *of a polygon* P *(the* pseudo-histogram partition of P*) and the partial cover of* P *by the SAM subpolygons* $SAMPC(\Pi)$

Lemma 2.3. $\|SAMPC(\Pi)\| \leq 8\,|P|$.

Proof. For any chord γ of $\widehat{\Gamma}(\Pi)$ and vertex v of $Q = \mathrm{vis}_\perp(\gamma)$ let $l(\gamma, v)$ be the line through v and perpendicular to γ and $V(\gamma, v)$ be the vertices of Q belonging to line $l(\gamma, v)$. If there are elements of $V(P) \cap V(\gamma, v)$ on both sides of γ then let $c(\gamma, v)$ be the number of elements of $V(\gamma, v)$ on the same side of γ as v; otherwise let $c(\gamma, v) = |V(\gamma, v)|$. Then $c(\gamma, v_\gamma) \leq 2$.

Let u be any vertex of P. Then, by Lemma 2.1, there exists at most 2 vertical elements of $\widehat{\Gamma}(\Pi)$, say γ_1 and γ_2, such that u is a vertex of $\mathrm{vis}_\perp(\gamma_1)$ and a vertex of $\mathrm{vis}_\perp(\gamma_2)$. Now $c(\gamma_1, u) + c(\gamma_2, u) \leq 4$. By applying these counting arguments to every vertex of P, we count $4|P|$ vertices in total. Applying the same arguments to the horizontal elements of $\widehat{\Gamma}(\Pi)$, counts another $4|P|$ vertices. The lemma now follows since we have counted each vertex of every element of Π at least once. □

The result of Lemma 2.3 is not tight. For a tighter bound see [1].

Lemma 2.4. *Given* Π*,* $SAMPC(\Pi)$ *can be constructed in* $O(|P|)$ *time.*

Proof. Let T_h and T_v be the horizontal trapezoidization and vertical trapezoidization of P respectively. Now, for each vertical chord γ_i of $\widehat{\Gamma}(\Pi)$, we construct $Q_i = \mathrm{vis}_\perp(\gamma_i)$ using the trapezoids of T_h contained in Q_i or intersecting Q_i. This is easily done in $O(|Q_i|)$ time. We must also delete duplicates and subpolygons contained in larger subpolygons but this is easily done in $O(|P|)$ time.[2] Next, we repeat this construction for the horizontal chords of $\widehat{\Gamma}(\Pi)$ using T_v. Thus, constructing $\mathrm{SAMPC}(P)$ requires $O(\|\mathrm{SAMPC}(P)\| + |T_h| + |T_v|) = O(|P|)$ time by Lemma 2.3. □

Clearly, from Observation 1.1, and Lemmas 1.2, 2.2, and 2.4, we can solve the LR problem on P in $O(n \log n)$ time by constructing a rectilinear tree partition of P in $O(n \log n)$ time in which the LR problem can be solved on the elements of the partition in $O(n \log n)$ time.

3 The LR Problem for Rectilinear Polygons

Let P be a polygon. We say that P is a *histogram polygon* if $P = \mathrm{vis}_\perp(\mathrm{base}(P))$. We point out that in [7] a histogram polygon is called a normal histogram polygon. We also point out that, unlike standard histograms, these polygons need not be rectilinear. We denote $\mathrm{hist}(P) = \mathrm{vis}_\perp(\mathrm{base}(P))$. Let γ be any axis-aligned chord of P and $Q = P_\gamma$. Note that $\mathrm{hist}(Q) = \mathrm{vis}_\perp(\mathrm{door}(Q), Q)$. We point out that any histogram polygon is a degenerate SAM polygon. We solve the LR problem for rectilinear polygons using histogram partitions; see [9].

Definition 3.1. [3] If $P = \mathrm{hist}(P)$ then

$$\pi_{\mathrm{hist}}(P) = \{P\}\,;$$

otherwise

$$\pi_{\mathrm{hist}}(P) = \{\mathrm{hist}(P)\} \cup \Bigg(\bigcup_{\gamma_i \in \mathrm{windows}(\mathrm{hist}(P))} \pi_{\mathrm{hist}}(P_{\gamma_i}) \Bigg)\,.$$

We refer to $\pi_{\mathrm{hist}}(P)$ as the *histogram partition* of P;

[2] By being more careful, we can avoid generating duplicates and subpolygons contained in other subpolygons in the first place.

[3] This definition is based on a similar definition in [9].

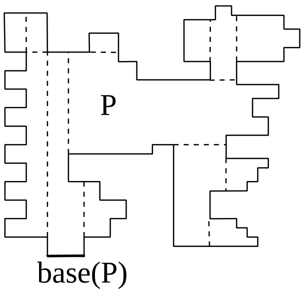

Figure 2: *A rectilinear polygon P and $\pi_{\text{hist}}(P)$*

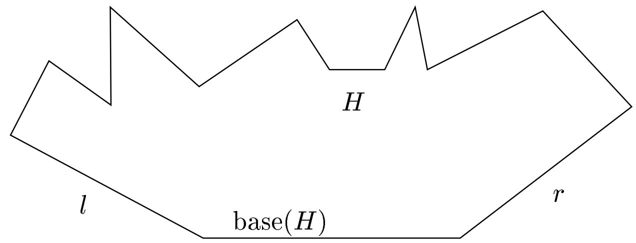

Figure 3: *A pseudo-histogram*

see Fig. 2. The histogram partition of a rectilinear polygon can be constructed in linear time [8]; see also [9].

Theorem 3.2. *The LR problem can be solved on an n vertex rectilinear polygon in $O(n \log n)$ time.*

Proof. Let Q be an n vertex rectilinear polygon. Let $\Pi = \text{SAMPC}(Q) \cup \text{hist}(Q)$. Now $\pi_{\text{hist}}(Q)$ is clearly a rectilinear tree partition and each element of $\pi_{\text{hist}}(Q)$ is a SAM polygon so $\text{SAMPC}(Q) \cup \pi_{\text{hist}}(Q)$ is a cover of Q by SAM polygons and furthermore, $\mathtt{C}_{\square}(\text{SAMPC}(Q) \cup \pi_{\text{hist}}(Q)) = 1$ by Lemma 2.2. However, this considers more subpolygons than necessary. We observe that every element of $\pi_{\text{hist}}(Q)$, except $\text{hist}(Q)$, is contained in an element of $\text{SAMPC}(Q)$. Thus, Π is also a cover of Q by SAM polygons and $\mathtt{C}_{\square}(\Pi) = 1$. Furthermore, Π can be constructed in $O(|Q|)$ time by Lemma 2.4. Thus, by Observation 1.1 and Lemma 1.2, we can solve the LR problem on Q in $O(n \log n)$ time. $\square$

4 Generalization to Simple Polygons

The same basic idea that we used to solve the LR problem for rectilinear polygons can be applied to solve the LR problem for simple polygons. However, we cannot use a partition of the input polygon into histograms for this problem because in general the amount of space required for such a partition could be unbounded; see [7]. Instead, we will use a partition of the input polygon into pseudo-histogram polygons [7].

Let H be a simple polygon, l be the edge immediately preceding base(H), and r be the edge immediately following base(H). We say that H is a *pseudo-histogram polygon* or just *pseudo-histogram* if the endpoints of base(H) are convex and, for any point p interior to H, there is point q on either base(H), l, or r such that directed line segment $\overrightarrow{pq}$ is perpendicular to base(H), interior to H, and directed towards the line through base(H). Let C be the subchain of the chain $l, \text{base}(H), r$ that is reachable from some interior point s of H by a directed line segment that is perpendicular to base(H), interior to H, and directed towards the line through base(H). We call C the *base chain* of H and denote it by baseChain (H). Note that, in degenerate cases, a pseudo-histogram polygon may have a base chain of length 1 or 2 but it must always have a base edge. See Fig. 3.

Let P be a polygon. A variation of the method used to partition polygons into histogram polygons can be used to partition polygons into pseudo-histograms. Let l be the edge of P preceding base(P) and r be the edge of P following base(P). Now let ζ be the chain of P that consists of:

- l if the shared vertex of base(P) and l is convex and its interior angle is obtuse,
- base(P), and
- r if the shared vertex of base(P) and r is convex and its interior angle is obtuse.

We denote by phist(P) the subpolygon Q of P such that, for any point p interior to Q, there is a point $q \in \zeta$ where line segment $\overline{pq}$ is perpendicular to base(P) and interior to P. Now we define $\pi_{\text{phist}}(P)$ analogously to the definition of $\pi_{\text{hist}}(P)$ except that we replace each instance of "hist" by "phist". We call $\pi_{\text{phist}}(P)$ the *pseudo-histogram partition* of P; see Fig. 1.

Partition $\pi_{\text{phist}}(P)$ can be constructed in $O(|P|)$ time [7]. Furthermore, $\mathtt{C}_{\square}(\text{SAMPC}(P) \cup \pi_{\text{phist}}(P)) = 1$ since $\pi_{\text{phist}}(P)$ is a rectilinear tree partition and by Lemma 2.2. However, unlike the rectilinear polygon case, we still need to find the largest rectangle in each element of $\pi_{\text{phist}}(P)$. The method we use is to construct covers of the pseudo-histograms by SAM polygons and quadrilaterals for which the rectilinear visibility cover number is 1. These covers are constructed in 2 stages. In the first stage we construct covers of the pseudo-histograms by SAM subpolygons and polygons we call nose histograms. In the second stage we construct covers of the nose histograms by SAM subpolygons and quadrilaterals. The set of SAM polygons and quadrilaterals constructed by this 2 stage process forms covers of the pseudo-histograms with a rectilinear visibility cover number of 1.

Let $H = \{v_0, \ldots, v_{m-1}\}$ be an m vertex polygon. We say that H is a nose polygon if edges $\overline{v_1 v_2}$ and $\overline{v_{m-1} v_0}$ are axis-aligned and perpendicular and, for some edge $e \in \{\overline{v_1 v_2}, \overline{v_{m-1} v_0}\}$, we have that, for every point p interior to H, there is a point, say q, on edge $\overline{v_0 v_1}$ such that the line segment $\overline{pq}$ is interior to H and parallel to e. We call $\overline{v_0 v_1}$ the *base edge* of H and chain $v_{m-1}v_0v_1v_2$ the *base chain* of H. Note that base(H) is not axis-aligned.

Let $S = \{v_0, \ldots, v_{m-1}\}$ be a pseudo-histogram and assume without loss of generality that base(S) is vertical and directed downward. We also assume that $|\text{baseChain}(S)| = 3$; the case that $|\text{baseChain}(S)| < 3$ is handled in a similar but simpler manner and is omitted. We construct a cover of S by nose histograms and SAM polygons as follows.

Let baseChain$(S) = stuv$ where $v_0 = t$. Let $\overline{rs}$ and $\overline{vw}$ be vertical chords of S; the degenerate and simpler cases that $\overline{rs}$ or $\overline{vw}$ or both don't exist are omitted. Let $\overline{tp}$ and $\overline{uq}$ be horizontal chords of S. If $\overline{rs} \cap \overline{tp} - \{r\} \neq \emptyset$ then let Q be the set containing the largest subpolygon of S containing both $\overline{rs}$ and $\overline{tp}$ as edges; otherwise let $Q = \emptyset$. If $\overline{vw} \cap \overline{uq} - \{w\} \neq \emptyset$ then let U be the set containing the largest subpolygon of S containing both $\overline{vw}$ and $\overline{uq}$ as edges; otherwise let $U = \emptyset$. We denote $\pi_{\text{noseSam}}(S) = \{\text{vis}_\perp(\overline{rs}), \text{vis}_\perp(\overline{tp}), \text{vis}_\perp(\overline{vw}), \text{vis}_\perp(\overline{uq})\}$ and $\pi_{\text{nose}}(S) = Q \cup U$. Let $\Pi = \pi_{\text{noseSam}}(S) \cup \pi_{\text{nose}}(S)$. Then it is easily established that $\mathtt{C}_\Box(\Pi \cup \text{hist}(S)) = 1$ and $\|\Pi\| = O(m)$. Furthermore, Π can be constructed $O(m)$ time. Now, since $\pi_{\text{nose}}(S)$ is empty or contains nose histograms and by Lemma 1.1, we have reduced the LR problem on pseudo-histograms to the LR problem on nose histograms or SAM polygons and furthermore, this reduction is linear. See Fig. 4.

We next reduce the LR problem on nose histograms to the LR problem on SAM polygons.[4] Let N be a nose histogram with base chain $tuvw$. For simplicity we assume that the edges of N other than $\overline{tu}$ and $\overline{vw}$ are not axis-aligned; it is not difficult to deal with additional axis-aligned edges and we omit the details. Assume without loss of generality that the start vertex is u and u is the top rightmost vertex of N. We will decompose N into a cover Π such that $\mathtt{C}_\Box(\Pi) = 1$ and Π contains only SAM polygons and quadrilaterals. Initially $\Pi = \emptyset$.

Let s be the point on edge $\overline{uv}$ such that $\overline{ws}$ is a horizontal chord of N. If w is a reflex vertex then let w' be the point on N such that $\overline{ww'}$ is a vertical chord of N; otherwise let $w' = w$. Let r be the destination vertex of the edge containing w'; if w' is a vertex of N then we choose the edge for which w' is the source vertex. For any point p we denote the x-coordinate of p by $x(p)$. See Fig. 5. There are 2 cases to consider:

1. $x(r) \leq x(s)$. Let r_h and r_ν be the points on edge $\overline{uv}$ such that $\overline{rr_h}$ is a horizontal chord of N and $\overline{rr_\nu}$ is a vertical chord of N. Then we add quadrilateral $vw'rr_h$ to Π and, if $w' \neq w$, we add polygon $\text{vis}_\perp\left(\overline{ww'}\right)$ to Π as well. Let N_1 be the subpolygon of N determined by chord $\overline{rr_\nu}$ and containing vertex u. Now, any rectangle, say R, in N must be contained in either quadrilateral $vw'rr_h$, polygon $\text{vis}_\perp\left(\overline{ww'}\right)$, or N_1. Otherwise R would need to intersect the area of overlap of $vw'rr_h$ and N_1 and then R would need to intersect both $\overline{rr_h}$ and $\overline{rr_\nu}$ which is impossible.

2. $x(r) > x(s)$. Let s_ν be the point on the subchain of H from w' to t such that $\overline{ss_\nu}$ is a vertical chord of N. We compute the SAM subpolygons $S_h = \text{vis}_\perp(\overline{ws})$ and $S_\nu = \text{vis}_\perp(\overline{ss_\nu})$ of N, and add them to Π. Let N_1 be the subpolygon of N determined by chord $\overline{ss_\nu}$ and containing vertex u. Now, any rectangle, say R', in N must be contained in one of S_h, S_ν, or N_1. Otherwise R' would need to intersect the area of overlap of S_h and S_ν or the overlap of S_ν and H_1. But, in the first case, R' would then need to intersect both $\overline{ss_\nu}$ and $\overline{ws}$, which is impossible. In the second case, R' would need to intersect both $\overline{ss_\nu}$ and the horizontal chord of H with an endpoint at s_ν, which is also impossible. Note that if $w' \neq w$ then $\text{vis}_\perp\left(\overline{ww'}\right)$ is contained in S_ν and thus is not put into Π.

Assume that N is represented by a doubly linked list of its vertices. Then, for either of the above cases, we can construct the subpolygons to put into Π and the subpolygon N_1 as a doubly linked list in $O(|N| - |N_1|)$ time and space.

Thus, we can repeat the above procedure until a SAM polygon is added to Π that has u has a vertex in $O(|N|)$ time and space. We denote Π by $\pi_{\text{samQuad}}(N)$.

Now, by an inductive argument, it is clear that any rectangle in N is contained in an element of Π. We denote

$$\pi_{\text{sam}}(S) = \pi_{\text{noseSam}}(S) \cup \Big(\bigcup_{Q_i \in \pi_{\text{nose}}(S)} \pi_{\text{samQuad}}(Q_i) \Big).$$

We summarize the above discussion with the following lemmas:

Lemma 4.1. $\pi_{\text{sam}}(S) \cup \text{hist}(S)$ *is a cover of* S *and* $\mathtt{C}_\Box(\pi_{\text{sam}}(S) \cup \text{hist}(S)) = 1$.

Lemma 4.2. $\pi_{\text{sam}}(S)$ *can be built in* $O(|S|)$ *time.*

[4] In fact this is is a poor idea in the sense that (we claim) the LR problem on a nose polygon can be solved in linear time but a better algorithm would not help as long as the cost solving the LR problem on an m vertex SAM polygon is $O(m \log m)$.

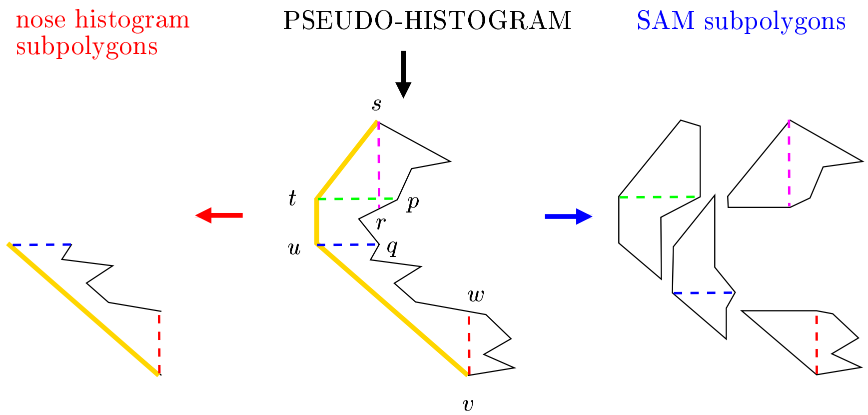

Figure 4: *A pseudo-histogram can be covered by at most 4 SAM polygons and 2 nose histogram polygons such that the rectilinear visibility cover number of the cover is 1*

Lemma 4.3. $\|\pi_{\mathrm{sam}}(S)\| \leq 9n$.

Proof. We consider only the case that S is a nose histogram. The more general case can be shown not to result is a larger cover. It is easily shown that the worst case scenario during the construction of $\pi_{\mathrm{sam}}(S)$ is that we add 2 SAM subpolygons to $\pi_{\mathrm{sam}}(S)$ but the size of the next nose histogram constructed has only one vertex less than the previous one. The subpolygons added to $\pi_{\mathrm{sam}}(S)$ in this situation can have as many as 9 vertices in total. □

4.1 A Cover with a Rectangle Visibility Cover Number of 1

In this subsection we construct a SAM subpolygon cover of an input polygon with rectilinear visibility cover number of 1. We denote

$$\begin{aligned}\pi_{\mathrm{sam2}}(P) &= \mathrm{SAMPC}(\pi_{\mathrm{phist}}(P)) \cup \mathrm{hist}(P) \\ &\cup \left(\bigcup_{Q_i \in \pi_{\mathrm{phist}}(P)} \pi_{\mathrm{sam}}(Q_i)\right).\end{aligned}$$

The key properties of $\pi_{\mathrm{sam2}}(P)$ are captured in the following lemmas:

Lemma 4.4. *$\pi_{\mathrm{sam2}}(P)$ is a cover of P and $\mathtt{C}_{\square}(\pi_{\mathrm{sam2}}(P)) = 1$.*

Proof. By construction and Lemma 2.2, if any rectangle, say R, in P is not contained in $\mathrm{SAMPC}(\pi_{\mathrm{phist}}(P)) \cup \mathrm{hist}(P)$ then it is contained in some element Q_i of $\pi_{\mathrm{phist}}(P)$ but not in $\mathrm{vis}_{\perp}(\mathrm{base}(Q_i))$. But then R is contained in $\pi_{\mathrm{sam}}(Q_i)$ by Lemma 4.1. □

Lemma 4.5. $\|\pi_{\mathrm{sam2}}(P)\| \leq 9|P|$.

Proof. See [1]. □

Lemma 4.6. *$\pi_{\mathrm{sam2}}(P)$ can be built in $O|P|$ time.*

Proof. This follows since $\pi_{\mathrm{phist}}(P)$ can be constructed in $|P|$ time and space and by Lemmas 2.4, 4.2, and 4.5. □

5 Conclusion

From Lemmas 4.4, 4.6, and 1.2 we get the following result:

Theorem 5.1. *Let P be an n vertex polygon. The LR problem can be solved on P in $O(n \log n)$ time.*

This is an improvement by a factor of $O(\log n)$ over the algorithm in [4]. We made this improvement by replacing one rectangle visibility cover by another more efficient one in that the new cover can be constructed in $O(\log n)$ less time and uses $O(\log n)$ less space. However, since it is not necessary to store all of the cover at once in order to solve the LR problem and the algorithm in [4] avoids doing so, there is no corresponding improvement in space efficiency.

It is also noteworthy that the only part of our algorithm that depends on the fact that we are searching for the largest rectangle according to its area is the method used to find the largest area rectangle in a SAM polygon. If we wanted to find the largest rectangle according to it perimeter, for example, then the algorithms presented here still work. We would only need to solve the problem in the special case of SAM polygons.

It is interesting to ask if we can improve upon the results of this paper by constructing a cover of a simple polygon into subpolygons simpler than SAM polygons such that the rectilinear cover number is 1. This is being explored. So far we have been able to do this in some special cases but not in the general case. We would also like to be able to construct a cover of a polygon with holes by SAM subpolygons, using linear space, such that the rectilinear cover number is 1.

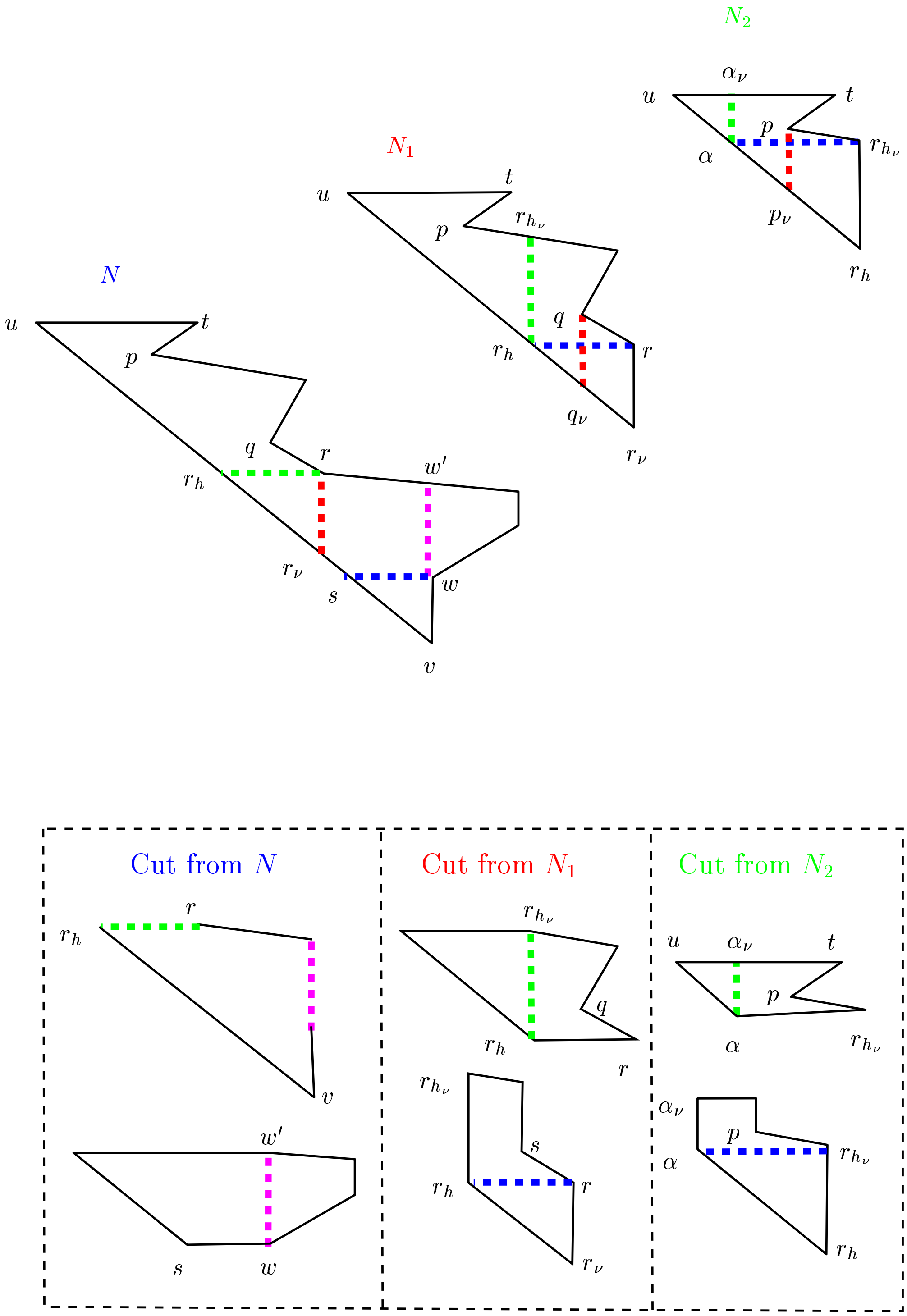

Figure 5: A cover of a nose histogram N by SAM polygons and quadrilaterals with a rectilinear cover number of 1

It is also interesting to ask if our methods can be used to find other kinds of subpolygons in a simple polygon. For example we could try to find the largest SAM polygon, the largest separated orthogonally convex polygon or the largest separated convex polygon. We believe that the first two problems are not difficult and we plan to add solutions to them to the final version of this paper. The last problem is clearly more difficult.

References

[1] R. P. Boland. *Polygon Visibility Decompositions with Applications.* Ph.D. thesis, submitted January, 2000, University of Ottawa., Ottawa, Ontario, Canada, 2000.

[2] Bernard Chazelle. Triangulating a simple polygon in linear time. *Discrete Comput. Geom.*, 6(5):485–524, 1991.

[3] Bernard Chazelle and Janet Incerpi. Triangulation and shape-complexity. *ACM Trans. Graph.*, 3(2):135–152, 1984.

[4] Karen Daniels, Victor Milenkovic, and Dan Roth. Finding the largest area axis-parallel rectangle in a polygon. *Comput. Geom.*, 7(1-2):125–148, 1997. Fifth Canadian Conference on Computational Geometry (Waterloo, ON, 1993).

[5] A. Fournier and D. Y. Montuno. Triangulating simple polygons and equivalent problems. *ACM Trans. Graph.*, 3(2):153–174, 1984.

[6] Martin Held. Efficient and reliable triangulation of polygons. In *Proc. Comput. Graphics Internat. 1998*, pages 633–643, June 1998.

[7] Rolf Klein and Andrzej Lingas. A linear-time randomized algorithm for the bounded Voronoi diagram of a simple polygon. In *Proc. 9th Annu. ACM Sympos. Comput. Geom.*, pages 124–132, 1993.

[8] C. Levcopoulos. *Heuristics for Minimum Decompositions of Polygons.* Ph.D. thesis, Linköping Studies in Science and Technology, Linköping, Sweden, 1987. No. 155.

[9] S. Schuierer. An optimal data structure for shortest rectilinear path queries in a simple rectilinear polygon. *Internat. J. Comput. Geom. Appl.*, 6:205–226, 1996.