Inscribing or Circumscribing a Histogon to a Convex Polygon

Jaehoon Chung
Department of Computer Science and Engineering, Pohang University of Science and Technology, Korea

Sang Won Bae
Division of Computer Science and Engineering, Kyonggi University, Suwon, Korea

Chan-Su Shin
Division of Computer Engineering, Hankuk University of Foreign Studies, Seoul, Korea

Sang Duk Yoon
Department of Service and Design Engineering, SungShin Women's University, Seoul, Korea

Hee-Kap Ahn
Graduate School of Artificial Intelligence, Department of Computer Science and Engineering, Pohang University of Science and Technology, Korea

Abstract

We consider two optimization problems of approximating a convex polygon, one by a largest inscribed histogon and the other by a smallest circumscribed histogon. An axis-aligned histogon is an axis-aligned rectilinear polygon such that every horizontal edge has an integer length. A histogon of orientation θ is a copy of an axis-aligned histogon rotated by θ in counterclockwise direction. The goal is to find a largest inscribed histogon and a smallest circumscribed histogon over all orientations in $[0, \pi)$. Depending on whether the horizontal width of a histogon is predetermined or not, we consider several different versions of the problem and present exact algorithms. These optimization problems belong to shape analysis, classification, and simplification, and they have applications in various cost-optimization problems.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Shape simplification, Shape analysis, Histogon, Convex polygon

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2022.13

Funding This research was partly supported by the Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2017-0-00905, Software Star Lab (Optimal Data Structure and Algorithmic Applications in Dynamic Geometric Environment)) and (No. 2019-0-01906, Artificial Intelligence Graduate School Program(POSTECH)).
Sang Won Bae: supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2018R1D1A1B07042755).
Chan-Su Shin: supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2019R1F1A1058963).
Sang Duk Yoon: supported by "Cooperative Research Program for Agriculture Science & Technology Development (Project No. PJ015269032022)" Rural Development Administration, Republic of Korea.

1 Introduction

We consider two optimization problems of approximating a convex polygon, one by a largest inscribed histogon and the other by a smallest circumscribed histogon. An axis-aligned *histogon* is an axis-aligned rectilinear (possibly weakly simple) polygon such that every

42nd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2022).
Editors: Anuj Dawar and Venkatesan Guruswami; Article No. 13; pp. 13:1–13:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

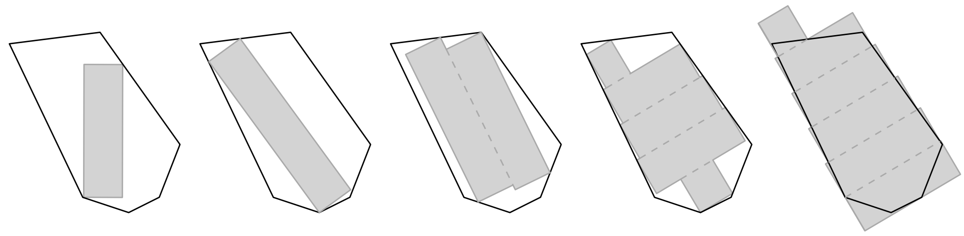

Figure 1 (a) The largest axis-aligned inscribed unit histogon. (b) The largest inscribed unit histogon. (c) The largest inscribed 2-histogon. (d) The largest inscribed histogon. (e) The smallest circumscribed histogon.

horizontal edge has an integer length. We call an axis-aligned histogon of width 1 an axis-aligned *unit histogon* and an axis-aligned histogon of *width* k an axis-aligned k-*histogon*. Thus, an axis-aligned unit histogon is simply an axis-aligned rectangle of horizontal width 1, and its height is the length of the vertical sides. An axis-aligned k-histogon for a positive integer k can be described by k axis-aligned and interior-disjoint unit histogons. A histogon of a fixed orientation $\theta \in [0, \pi)$ is a copy of an axis-aligned histogon rotated by θ in counterclockwise direction.

In the inscribed histogon problem, we compute a histogon with maximum area that can be inscribed in P. We call such a histogon a *largest* inscribed histogon of P. Depending on whether the horizontal width of a histogon is predetermined (1 or a positive integer k) or not, we consider three versions of the problem. In the circumscribed histogon problem, we compute a histogon with minimum area that can be circumscribed to P. We call such a histogon a *smallest* circumscribed histogon of P. See Figure 1 for an illustration.

The optimization problems we investigate belong to shape analysis, classification, and simplification [2, 3]. Many optimization problems occurred in those research topics are concerned with the largest inscribed figure and the smallest circumscribed figure of a prescribed shape. The largest inscribed histogon problem and the smallest circumscribed histogon problem have applications in several coverage path planning (CPP) problems [7, 11, 15, 16] such as mowing a lawn using a mower, painting a piece using a spray gun, inspecting the surface of an object by a scanner, and milling a pocket by moving a cutter. Other applications include several topics in calculus, including Riemann sums and optimization. For a function graph (or a curve), the area under the graph can be approximated by a histogon: an inscribed histogon is an under-approximation of the area and a circumscribed histogon is an over-approximation of the area.

Related Work. There has been a lot of work in approximating shapes in past decades. The goal is to find a polygon inscribing or circumscribing another polygon (a convex polygon or a simple polygon) while maximizing (or minimizing) a certain measure. There are algorithms for finding triangles with maximum area or maximum perimeter inscribed in a convex polygon and a simple polygon [20, 24]. A convex k-gon with maximum area or maximum perimeter inscribed in a convex n-gon can be computed in $O(kn + n \log n)$ time [1, 24]. Chang et al. [8] gave an $O(n^7)$-time algorithm for finding a convex polygon with maximum area and an $O(n^6)$-time algorithm for finding a convex polygon with maximum perimeter inscribed in a convex n-gon. There are $O(n)$-time algorithms for finding triangles with minimum area or minimum-perimeter circumscribing a convex n-gon [5, 21].

DePano et al. [14] gave algorithms for finding an equilateral triangle with maximum area and a square with maximum area inscribed in a polygon either convex or simple. Lee et al. [19] gave algorithms for finding maximum-area triangles with fixed interior angles inscribed in a polygon, either convex, simple, or even non-simple possibly with holes. Intensive research has been done for the problems of finding rectangles with maximum area or maximum perimeter inscribed in a convex polygon and a simple polygon [6, 10, 18]. Jin et al. [17] gave an $O(n^2)$-time algorithm for computing all parallelograms with maximum area in a convex n-gon. Toussaint [23] gave an $O(n)$-time algorithm for finding rectangles with minimum area or minimum perimeter circumscribing a convex n-gon. Schwarz et al. [22] gave a simple $O(n)$-time algorithm for finding a parallelogram with minimum area in a convex n-gon.

Very recently, the authors [12] presented first algorithms for the axis-aligned case of our histogon problem for a convex polygon P. These previous algorithms include: an $O(\log n)$-time algorithm for a largest axis-aligned inscribed unit histogon, an $O(\min\{n, k \log^2 \frac{n}{k}\})$-time algorithm for a largest axis-aligned inscribed k-histogon for a fixed $k > 1$, an $O(\min\{n, w \log^2 \frac{n}{w}\})$-time algorithm for a largest axis-aligned inscribed histogon, where w denotes the width of a largest axis-aligned histogon inscribed in P, and an $O(\min\{n, W \log \frac{n}{W}\})$-time algorithm for a smallest axis-aligned circumscribed histogon, where W denotes the horizontal width of P.

Our Results. For the problem of inscribing a largest unit histogon in a convex n-gon, we present an $O(n \log n + U)$-time algorithm using $O(n)$ space, where U denotes the total number of intersections between unit circles centered at vertices of P and the edges of P. In the worst case, the quantity U can be quadratic in n, while it is near-linear in most cases. In addition, we present another algorithm that runs in $O(n \log n + n/\delta)$ time under the assumption that there exists a unit histogon of height at least δ in P where $0 < \delta < 1$. This provides a faster way once one asserts the existence of any unit histogon of positive height contained in P, by any means such as efficient approximation algorithms. We also show that our algorithm can determine whether there exists a unit histogon of height δ in P in the same time bound for any input $0 < \delta < 1$. Based on these results, a largest inscribed unit histogon can be computed more efficiently: in $O(n \log n)$ time if its height h is $\Omega(1/\log n)$, or in $O(n \log n \log\log \frac{1}{h \log n} + n/h^2)$ time in an output-sensitive way, otherwise.

We also present an algorithm that, given a positive integer k, finds a largest k-histogon inscribed in P in $O(kn^2(\log n + kT(\min\{k, n\})))$ time using $O(\min\{k, n\}n)$ space, where $T(m)$ denotes the time complexity of the optimization step for the trigonometric expression with $O(m)$ terms, each of quadratic form. For finding a largest histogon with no restriction on the width inscribed in P, we present an $O(Dn^2(\log n + T(\min\{D, n\})))$-time algorithm using $O(\min\{D, n\}n)$ space, where D denotes the diameter of P. Barequet and Rogol [4] observed that $T(m) = O(m)$ in practice.

Finally, for finding a smallest circumscribed histogon of a convex n-gon, we present an $O(Dn(\log(\min\{D, n\}) + T(\min\{D, n\})))$-time algorithm using $O(n)$ space.

Due to lack of space, some proofs and details are omitted. They can be found in the full version of the paper.

2 Preliminaries

Let P be a convex polygon with n vertices, given in a list sorted in counterclockwise order along the boundary of P. For ease of discussion, we assume that no two edges of P are parallel to each other.

For a connected set X, we denote by ∂X the boundary of X, by $\mathsf{int}(X)$ the interior of X, and by $\mathsf{cl}(X)$ the closure of X. For a point $p \in \mathbb{R}^2$, let $x(p)$ and $y(p)$ be the x-coordinate and the y-coordinate of p, respectively. For any two points p and q in $\mathbb{R}^2$, we use pq to denote the line segment connecting p and q, and by $|pq|$ the length of pq. If both p and q lie on ∂P, pq is called a *chord* of P. A chord of unit length in P is called *a unit chord* of P. We use $\partial P[x, y]$ to denote the portion of ∂P from x to y in counterclockwise order, and let $\partial P(x, y] = \partial P[x, y] \setminus \{x\}$, $\partial P[x, y) = \partial P[x, y] \setminus \{y\}$, and $\partial P(x, y) = \partial P[x, y] \setminus \{x, y\}$. We use $P[x, y]$ to denote the subpolygon of P enclosed by $\partial P[x, y]$ and the chord xy.

A histogon of a fixed orientation $\theta \in [0, \pi)$ is a copy of an axis-aligned histogon rotated by θ in counterclockwise direction. The width of a histogon H of orientation θ is the width of the axis-aligned copy of H that is obtained by rotating H by θ in clockwise direction. Let $w(H)$ be the width of H and $|H|$ denote the area of H.

The *orientation* of a line is the angle swept from the x-axis in a counterclockwise direction to the line, and it is thus in $[0, \pi)$. The *orientation* of a line segment is that of the line containing it. We mean by a *boundary element* of P its vertex or edge. For each $\theta \in [0, \pi)$, there are exactly two lines of orientation θ that are tangent to P. Each of these two tangent lines intersects ∂P in a boundary element of P. We call a pair (m_1, m_2) of boundary elements of P *antipodal* if there exists an orientation $\theta \in [0, \pi)$ of which two tangent lines intersect P in m_1 and m_2. Toussaint [23] showed the following, introducing the rotating caliper.

▶ **Lemma 1** (Toussaint [23]). *There are $O(n)$ antipodal pairs of a convex n-gon P, and they can be computed in $O(n)$ time.*

Note that if (m_1, m_2) is an antipodal pair, then not both of m_1 and m_2 can be edges of P, since P has no two parallel edges.

3 Largest inscribed unit histogon

In this section we compute a largest inscribed unit histogon of P. Cabello et al. [6] gave an algorithm that computes a largest inscribed rectangle in a convex polygon with n vertices in $O(n^3)$ time. They showed that the set of parallelograms contained in a convex polygon with n vertices can be parameterized by a convex polytope in $\mathbb{R}^6$ defined by $4n$ linear constraints. Their algorithm triangulates the boundary of the convex polytope and finds a largest rectangle for each simplex of the triangulation. Since the optimization problem for a simplex has a constant size and the complexity of the convex polytope is $O(n^3)$, their algorithm takes $O(n^3)$ time.

We can modify their algorithm so that it works for our problem. For each of $O(n^3)$ optimization problems, we add a new non-linear constraint that a side of parallelogram has length 1. Since the size of altered optimization problem is constant, we can find a largest inscribed unit histogon of P in $O(n^3)$ time.

In the following, we present an algorithm for the problem that runs in $O(n \log n + U)$ time, where U denotes the total number of intersections between unit circles centered at vertices of P and the edges of P. Since a unit circle intersects an edge of P at most twice and it may intersect $O(n)$ edges of P, we have $U = O(n^2)$. Indeed, we can construct a convex polygon such that $U = \Omega(n^2)$. In practical situations, however, most unit circles centered at vertices intersect only few edges of P, so the total number of intersections is either linear or near-linear and our algorithm runs faster.

A largest inscribed unit histogon of P touches some boundary elements (edges and vertices) of P. For a unit histogon $\bar{H}$, we say that there is a *side-contact* if a side of $\bar{H}$ is fully contained in an edge of P, or a *corner-contact* if a corner of $\bar{H}$ lies on ∂P. The *contact set* of $\bar{H}$ is the set of all of its corner-contacts and side-contacts. We show that the contact set of any largest inscribed unit histogon falls into one of the following four types. See Figure 2.

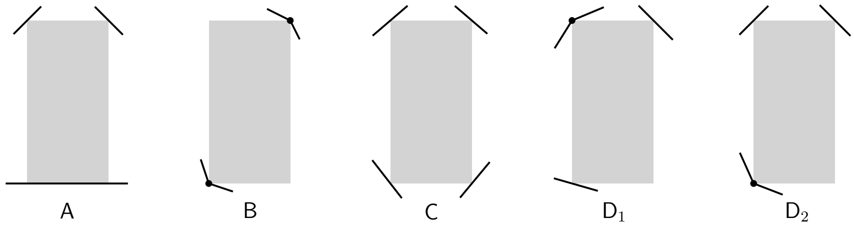

Figure 2 Types of contact sets of for largest inscribed unit histogons of P.

▶ **Lemma 2.** *For any largest inscribed unit histogon of P, its contact set satisfies one of the following conditions:*

A. *It has a side-contact.*

B. *It consists of two corner-contacts at two vertices of P that are antipodal.*

C. *It consists of four corner-contacts.*

D. *It consists of three corner-contacts, one of which is associated with a vertex of P.*

By Lemma 2, our algorithm finds a largest unit histogon contained in P whose contact set falls into each of the four types.

3.1 One side-contact (type A) or two corner-contacts (type B)

Any unit histogon of type A contained in P has a side-contact with an edge of P, so a side of such a histogon has the same orientation with an edge of P. This reduces the problem to its fixed-orientation variant, which can be solved in $O(\log n)$ time [12]. Hence, a largest inscribed unit histogon of type A can be found in $O(n \log n)$ time by solving $O(n)$ instances of the fixed-orientation problem.

For unit histogons of type B, we first specify all antipodal pairs of P by Lemma 1, and consider each antipodal pair (v_1, v_2) such that both v_1 and v_2 are vertices of P. There are at most two possible unit histogons H such that v_1 and v_2 are opposite corners of H. We try each of these two unit histogons and test if it is contained in P. This containment test can be done in $O(\log n)$ time [13]. Since there are $O(n)$ antipodal pairs by Lemma 1, we can find a largest inscribed unit histogon of type B in $O(n \log n)$ time.

▶ **Lemma 3.** *We can compute a largest inscribed unit histogon of P that has a side-contact (type A) or two corner-contacts (type B) in $O(n \log n)$ time.*

3.2 Four corner-contacts (type C)

A unit histogon of type C has top and bottom sides lying on two parallel unit chords of P by Lemma 2. We say two parallel unit chords of P are *aligned orthogonally* if their convex hull forms a unit histogon. A largest inscribed unit histogon of type C in P has a positive height if and only if there are the two distinct parallel unit chords of P that are aligned orthogonally.

We find all inscribed unit histogons of type C in P and return the largest one among them. To do this, we trace two unit chords of orientation θ in P while θ increasing from 0 to π, and find the orientations at which the two unit chords are aligned orthogonally. The following lemma is about the existence of two distinct parallel unit chords in P.

▶ **Lemma 4.** *The following statements are equivalent:*

- *The length of a longest chord of orientation θ in P is larger than 1.*
- *Either there are exactly two distinct unit chords of orientation θ, or P has an edge of orientation θ with length larger than 1.*

Let O denote the set of all orientations $\theta \in [0, \pi)$ such that the longest chord of orientation θ in P has length larger than 1, but there is no edge of P of orientation θ whose length is larger than 1. By Lemma 4, there are exactly two distinct unit chords of orientation θ in P if and only if $\theta \in O$. Note that O is our search space for all possible unit histogons of type C. In the following, we investigate the motion of the longest chord and unit chords of orientation θ as θ continuously increases.

▶ **Lemma 5.** *Both endpoints of the longest chord of orientation θ in P move monotonously in the counterclockwise direction along ∂P as θ increases. Moreover, the pair of two boundary elements of P on which the endpoints of the longest chord lie is antipodal.*

For any orientation $\theta \in [0, \pi)$, consider the longest chord ab of orientation θ and the two boundary elements e_a, e_b of P such that $a \in e_a$ and $b \in e_b$. It is obvious that the longest chord continuously moves as θ increases while its endpoints lie on e_a and e_b, respectively, unless both e_a and e_b are vertices. This implies that the length of the longest chord also changes continuously locally and thus that the set O forms several open intervals of orientations. Let $\mathcal{I}(O)$ be the set of these intervals induced by O.

By Lemma 5, together with Lemma 1, there are only $O(n)$ different pairs of such boundary elements (e_a, e_b) that the endpoints of the longest chord may land on. Hence, the set $\mathcal{I}(O)$ consists of $O(n)$ open intervals. These intervals can be computed in $O(n)$ time by processing each antipodal pair in $O(1)$ time after specifying them explicitly in $O(n)$ time by Lemma 1.

▶ **Lemma 6.** *Both endpoints of a unit chord of orientation θ move continuously along ∂P as θ continuously increases over any interval in $\mathcal{I}(O)$.*

We define the combinatorial structure of a unit chord to be the pair of the boundary elements of P where its endpoints lie. The combinatorial structures of unit chords of orientation θ may change for θ increasing in an interval of $\mathcal{I}(O)$. Since the endpoints of a unit chord move continuously along ∂P by Lemma 6, such a change occurs only if an endpoint of the unit chord meets a vertex of P. We call an orientation at which the combinatorial structure of a unit chord changes a *v-event* orientation. The set of v-event orientations partitions the intervals of $\mathcal{I}(O)$ into subintervals, called *v-intervals*. Note that, for any v-interval I, the combinatorial structure of the unit chords of orientation θ remains the same over all $\theta \in I$. Moreover, the number of v-intervals of $\mathcal{I}(O)$ is bounded by $O(n + U)$ since each v-event corresponds to an intersection between a unit disk centered at a vertex of P and an edge of P.

Now, we compute v-event orientations, v-intervals, and unit histogons of type C by processing the intervals of $\mathcal{I}(O)$, one by one, in the increasing order of orientation as follows. Consider an interval $I = (\theta_0, \theta_1) \in \mathcal{I}(O)$, and assume that we have processed all intervals prior to I and are now about to process I. By definition of O, note that $\theta_0 \notin O$. This implies that either there is an edge e_0 of P of orientation θ_0 whose length is larger than 1, or the longest chord of orientation θ_0 has length at most 1. For the former case, there is the previous interval $I' \in \mathcal{I}(O)$ that share the endpoint θ_0 with I, and thus we have the unit chord of orientation θ_0 not lying on e_0 when we process I. For the latter case, the length of the longest chord of orientation θ_0 is exactly 1 by Lemma 6. Thus, the longest chord is the only unit chord of orientation θ_0 and it can be specified when we compute $\mathcal{I}(O)$. As θ increases from θ_0 to θ_1 over I, the endpoints of the unit chords of orientation θ move continuously along ∂P for θ increasing in I by Lemma 6. So, we can trace the endpoints of the unit chords of orientation θ and compute v-event orientations in I and v-intervals induced by the orientations in order.

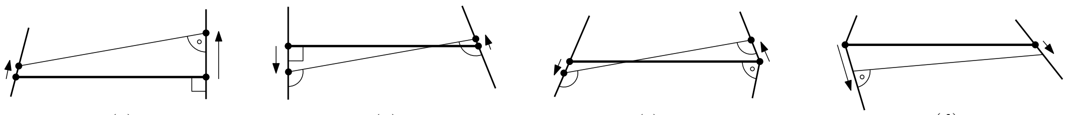

Figure 3 Two edge events and two vertex events where the feasibility of a top unit chord C (thick segment) changes for θ increasing from θ_1 in the interior of an interval of $\mathcal{I}(O)$. Assume $\theta_1 = 0$. Each acute angle is marked with a circle. (a) An edge event where C becomes infeasible. (b) An edge event where C becomes feasible. (c) A vertex event where C becomes feasible. (d) a vertex event where C becomes infeasible.

For a v-interval I' determined during this tracing, we check whether there is an orientation $\theta \in I'$ such that the two unit chords are aligned orthogonally, that is, the convex hull of the two unit chords of orientation θ forms a unit histogon. Since the combinatorial structures of the unit chords does not change in the v-interval, this can be done by solving an equation defined for $\theta \in I'$ such that the equation has a solution if and only if the two unit chords are aligned orthogonally. From the edges of P corresponding to the combinatorial structure of a unit chord of orientation $\theta \in I$, we can express the x-coordinate of an endpoint of the chord as $A\sin(\theta + B) + C$ using the law of sines, where A, B and C are constants. The equation has $O(1)$ solutions which can be found in $O(1)$ time.

To sum up, we compute all v-event orientations and v-intervals in $O(n + U)$ time. For each of the $O(n + U)$ v-intervals, we determine whether there is a unit histogon of type C of an orientation in the v-interval in $O(1)$ time. Thus, we compute all unit histogons of type C in $O(n + U)$ time.

▶ **Lemma 7.** *We can compute the largest inscribed unit histogon of P that has four corner-contacts (type C) in $O(n + U)$ time using $O(n)$ space.*

3.3 Three corner-contacts (type D)

We denote by $\bar{H}^*$ the largest inscribed unit histogon of type D in P and let θ^* be the orientation of its top and bottom sides. By Lemma 2, one corner of $\bar{H}^*$ lies at a vertex of P and the top or bottom side of $\bar{H}^*$ are unit chords of orientation θ^* in P.

We say a unit chord C of orientation $\theta \in O$ is *feasible* if there exists a unit histogon $\bar{H}$ in P of a positive height such that C is a top or bottom side of $\bar{H}$. If C is the top side (or the bottom side, respectively) of $\bar{H}$, we call C a *top* (or a *bottom*, respectively) unit chord and the orientation θ *top feasible* (or *bottom feasible*, respectively).

In the following, we consider the case that the top side of $\bar{H}^*$ is a top feasible unit chord and the bottom-left corner of $\bar{H}^*$ lies on ∂P. The other cases in which the bottom-right corner of $\bar{H}^*$ lies on ∂P or the bottom side of $\bar{H}^*$ is a bottom feasible unit chord can be handled analogously. Our strategy is to trace the top feasible unit chord of orientation θ while θ increases in each interval of $\mathcal{I}(O)$ and to find all unit histogons inscribed in P whose top side coincides with a top unit chord.

Events and event orientations. As θ continuously increases over an interval $I \in \mathcal{I}(O)$, a unit chord of orientation θ becomes feasible and infeasible at certain orientations in I. We call each such orientation an *f-event* orientation. There are two types of f-event orientations of a unit chord C. See Figure 3 for an illustration.

- *Edge* event: C is orthogonal to an edge on which an endpoint of C lies.
- *Vertex* event: An endpoint of C meets a vertex and both interior angles at the vertex of the two subpolygons of P induced by C are acute.

Then there is some $\zeta > 0$ such that either C is infeasible at $\theta - \epsilon$ but feasible at $\theta + \epsilon$ for any $0 < \epsilon \leq \zeta$, or C is feasible at $\theta - \epsilon$ but infeasible at $\theta + \epsilon$ for any $0 < \epsilon \leq \zeta$.

Note that we can determine whether a unit chord is at f-event or v-event (defined in Section 3.2) while tracing the endpoints of the unit chord. Thus, we can compute all f-event orientations along with v-event orientations while tracing the two unit chords.

For a top feasible orientation θ, we denote by $\alpha(\theta)$ the top unit chord of orientation θ and let $\bar{H}(\theta)$ be the largest unit histogon inscribed in P whose top side is $\alpha(\theta)$. Let $\alpha_1(\theta)$ and $\alpha_2(\theta)$ be the two endpoints of $\alpha(\theta)$ that correspond to the top-right corner and the top-left corner of $\bar{H}(\theta)$, respectively.

By Lemma 6, $\alpha_1(\theta)$ and $\alpha_2(\theta)$ move continuously along ∂P as θ continuously increases over the interval. Since P is convex, the endpoints of the two chords orthogonal to $\alpha(\theta)$ through $\alpha_1(\theta)$ and $\alpha_2(\theta)$ also move continuously along ∂P. Observe that $\bar{H}(\theta)$ always has at least one bottom corner at one endpoint of the orthogonal chords. When $\bar{H}(\theta)$ has both bottom corners at endpoints of the orthogonal chords, it has four corner-contacts (type C), and $\bar{H}(\theta - \varepsilon)$ has its bottom-left corner lying on ∂P and $\bar{H}(\theta + \varepsilon)$ has its bottom-right corner lying on ∂P (or in the opposite way) for sufficiently small ε.

Intervals containing no event orientations. We partition the intervals of $\mathcal{I}(O)$ by the f-event orientations, v-event orientations, and the orientations where unit histogons of type C are defined. Then we gather (partitioned) intervals I such that for any orientation $\theta \in I$, θ is a top feasible orientation and $\bar{H}(\theta)$ has its bottom-left corner $q(\theta)$ on ∂P. The resulting set of intervals is denoted by $\mathcal{I}_L$. The number of intervals in $\mathcal{I}_L$ is $O(n + U)$, and for an interval $I \in \mathcal{I}_L$, the combinatorial structure of $\alpha(\theta)$ remains the same for any orientation $\theta \in I$.

Let u denote the topmost vertex of P. If there are more than one topmost vertex, let u be the one with the largest x-coordinate. We claim that for any $I \in \mathcal{I}_L$ and $\theta \in I$, u, $\alpha_2(\theta)$ and $q(\theta)$ appear in counterclockwise order along ∂P. When $\theta = 0$, it clearly holds. Suppose that u lies on $\partial P[\alpha_2(\theta), q(\theta)]$ for some $\theta > 0$. Since both interior angles, one at $\alpha_2(\theta)$ and one at $q(\theta)$, in $P[\alpha_2(\theta), q(\theta)]$ are smaller than $\pi/2$, the line of orientation θ passing through u intersects $\partial P(q(\theta), \alpha_2(\theta))$ at a boundary point with y-coordinate larger than that of u, a contradiction.

▶ **Lemma 8.** *For any $I_1, I_2 \in \mathcal{I}_L$ and any two orientations $\theta_1 \in I_1$ and $\theta_2 \in I_2$ with $\theta_1 < \theta_2$, if $q(\theta_2) \in \partial P(u, q(\theta_1)]$, then $|\bar{H}(\theta_2)| < |\bar{H}^*|$.*

For any unit histogon $\bar{H}$ of type D, we distinguish two further subcases: $\bar{H}$ is of type D_1 if a corner incident to its top side lies on a vertex of P, or of type D_2 if its bottom-left corner lies on a vertex of P. For an illustration, see Figure 2. If $\bar{H}^*$ is of type D_1, θ^* is an endpoint of an interval of $\mathcal{I}_L$ by definition of v-event in Section 3.2. If $\bar{H}^*$ is of type D_2, $q(\theta^*)$ lies on a vertex of P. By Lemma 8, there is no $\theta \in I$ with $I \in \mathcal{I}_L$ such that $\theta < \theta^*$ and $q(\theta^*) \in \partial P(u, q(\theta)]$.

Algorithm. Our algorithm processes the intervals of $\mathcal{I}_L$ one by one in increasing order of orientation, and computes $\bar{H}^*$ as follows. It maintains a point w lying on ∂P to indicate the portion $\partial P[u, w]$ that has been processed so far. In the beginning, w is set to the topmost vertex u. Let $v(w)$ denote the counterclockwise neighbor vertex of w. The algorithm processes an interval $I \in \mathcal{I}_L$ as follows. Let θ_0 and θ_1 be the endpoints of I with $\theta_0 < \theta_1$. A largest inscribed unit histogon of type D_1 is computed in Step 1 and Step 3, and a largest inscribed unit histogon of type D_2 is computed in Step 2.

Step 1. If θ_0 is a top feasible orientation and $q(\theta_0) \in \partial P[w, u)$, compute $\bar{H}(\theta_0)$ and update w to $q(\theta_0)$.

Step 2. Repeat the following if there is $\theta \in [\theta_0, \theta_1]$ such that $q(\theta) \in \partial P[v(w), u)$.

a. Find the smallest $\theta' \in [\theta_0, \theta_1]$ such that $v(w)$ is met by $q(\theta')$, and compute $\bar{H}(\theta')$ if such θ' exists.

b. Update w to $v(w)$.

Step 3. If θ_1 is a top feasible orientation and $q(\theta_1) \in \partial P[w, u)$, compute $\bar{H}(\theta_1)$ and update w to $q(\theta_1)$.

▶ **Lemma 9.** *Our algorithm computes $\bar{H}^*$ while processing the interval I of $\mathcal{I}_L$ with $\theta^* \in I$.*

Analysis. Now we analyze the running time of the algorithm. The algorithm processes the intervals in $\mathcal{I}_L$ in the increasing order, one by one. Recall that it computes f-event orientations, v-event orientations, and orientations of unit histogons of type C while tracing the unit chords of orientation θ as θ runs over each interval in $\mathcal{I}(O)$. They can be found in the increasing order of orientation in $O(n + U)$ time using $O(n)$ space.

In Step 1, the algorithm checks whether $q(\theta_0) \in \partial P[w, u)$ or not in $O(1)$ time using $\alpha(\theta_0)$ and w. If $q(\theta_0) \in \partial P[w, u)$, it finds the edge where $q(\theta_0)$ lies by checking the edges of P one by one in counterclockwise order from the edge where w lies, and updates w to $q(\theta_0)$. Thus, Step 1 is done in time linear to the number of edges checked for updating w. Similarly, Step 3 can be done in the same time.

Now consider Step 2. The algorithm checks if there is an orientation $\theta \in [\theta_0, \theta_1]$ such that $q(\theta) \in \partial P[v(w), u)$. This can be done in $O(1)$ time as the combinatorial structure of $\alpha(\theta)$ has size $O(1)$ for $\theta \in [\theta_0, \theta_1]$. If there exists such an orientation, the algorithm finds the smallest $\theta' \in [\theta_0, \theta_1]$ such that $v(w)$ is met by $q(\theta')$, and updates w to $v(w)$. This also takes $O(1)$ time as the corresponding combinatorial structure has size $O(1)$. Then Step 2 can be done in time linear to the number of vertices checked for updating w. Note that w is updated to a point in $\partial P[w, u)$, and then the edges and vertices we check to update w change monotonically on ∂P in counterclockwise direction. Thus, the algorithm checks the edges and vertices of P for updating w in $O(n)$ time plus the time linear to the number of intervals in $\mathcal{I}_L$. Since there are $O(n+U)$ intervals in $\mathcal{I}_L$, the algorithm finds $\bar{H}^*$ in $O(n+U)$ time using $O(n)$ space.

▶ **Lemma 10.** *We can compute the largest inscribed unit histogon of P that has three corner-contacts (type D) in $O(n + U)$ time using $O(n)$ space.*

We have shown how to find the largest inscribed unit histogons in P for types A, B, C and D. Taken together, we choose the largest one among them as the largest inscribed unit histogon in P. From Lemmas 3, 7, and 10, we have the following theorem.

▶ **Theorem 11.** *We can determine whether there is an inscribed unit histogon with positive height in P in $O(n)$ time. If exists, we can find a largest inscribed unit histogon in P in $O(n \log n + U)$ time using $O(n)$ space, where U is the number of intersections between the unit disks centered at the vertices of P and the edges of P.*

▶ Remark. One may wonder whether there are convex polygons with n vertices for which the number of v-intervals is $\Omega(n^2)$. We show how to construct such a convex polygon in Figure 4. The polygon has roughly $\lfloor \frac{n}{2} \rfloor$ vertices that are very close to each other and roughly $\lfloor \frac{n}{2} \rfloor$ edges each of which contains a point at distance 1 from those vertices as shown in the figure. Thus, there are $\Omega(n^2)$ feasible unit chords with different combinatorial structures.

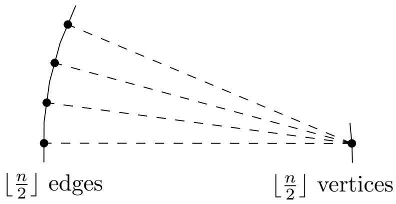

Figure 4 There can be $\Omega(n^2)$ v-event orientations.

4 Improved algorithms for a largest inscribed unit histogon

In this section, we assume that there exists a unit histogon of height δ inscribed in P for some $0 < \delta < 1$. Under this assumption, we show that a largest inscribed unit histogon in P can be computed in $O(n \log n + n/\delta)$ time, independent of the quantity U.

First, we show that there are $O(n)$ f-event orientations in total while θ increases from 0 to π, and that all f-event orientations can be computed in $O(n \log n)$ time.

▶ **Lemma 12.** *There are $O(n)$ f-event orientations in total.*

Proof. There are at most two unit chords that are orthogonal to each edge of P. Thus there are $O(n)$ edge events.

Now we count the number of vertex events on a vertex v. Suppose that there are three vertex events on v, induced by unit chords C_1, C_2 and C_3 in increasing order of orientation. By definition, the orientations of C_1, C_2 and C_3 are all in O. Observe that both interior angles at v of the subpolygons of P induced by C_1 are acute. The same holds for C_2 and C_3.

Let P_j be the subpolygon of P induced by C_2 that contains C_j for $j = 1, 3$. Let ℓ be the line containing the edge where the endpoint of C_2, other than its endpoint lying on v, lies. See Figure 5(a). Then ℓ does not intersect the interior of C_3. Since the isosceles triangle with two sides C_2 and C_3 has an acute angle at its base corners, the sum of the interior angles at the endpoints of C_2 in P_1 ($\phi_1 + \phi_2$ in Figure 5(a)) is strictly smaller than π. Similarly the sum of the interior angles at the endpoints of C_2 in P_3 is strictly smaller than π. Then C_2 is the longest chord contained in P at the orientation of C_2. This contradicts that the orientation of C_2 is in O. Recall that an orientation θ is an element of O if the longest chord of orientation θ in P has length larger than 1. Thus there are at most two vertex events on a vertex and there are $O(n)$ vertex events in total. ◀

We show how to find all $O(n)$ f-events in $O(n \log n)$ time. The authors in the previous

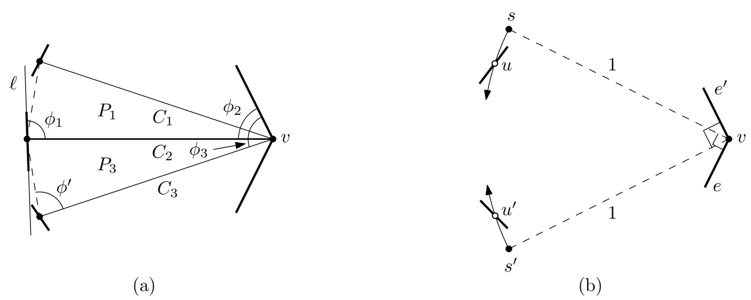

Figure 5 (a) $\phi' < \pi/2$ and $\phi_3 < \pi/2$ (acute). Since ℓ does not intersect the interior of C_3, $\phi' + \phi_1 \leq \pi = \phi_3 - \phi_2 + 2\phi'$. $\phi_1 + \phi_2 \leq \phi' + \phi_3 < \pi$. (b) Circular arc queries for finding vertex event orientations.

paper [12] showed that two unit chords of a fixed orientation can be found in $O(\log n)$ time by binary search using the sorted list of vertices of P. For each edge of P, we can find all unit chords which are incident and orthogonal to the edge in $O(\log n)$ time. Thus we can find all edge events in $O(n \log n)$ time.

There are at most two vertex events on a vertex as shown in the proof of Lemma 12. To find them, we construct a data structure supporting circular ray shooting queries on P such that for a directed query arc of a circle with a start point and a direction (clockwise or counterclockwise), it finds the first intersection between the arc and ∂P. Cheng et al. [9] gave a hierarchical decomposition of a simple polygon for circular ray shooting queries of a fixed radius with $O(n \log n)$ construction time and $O(n)$ space that supports $O(\log n)$ query time. For a vertex v of P, let e and e' be the edges incident to v such that e' is the counterclockwise neighbor of v. For any line tangent to P at v, let s and s' be the points lying in the side of the tangent line containing P such that they are at distance 1 from v and the segments sv and $s'v$ are orthogonal to e and e', respectively. See Figure 5(b) for an illustration. We perform two circular arc queries, one with the counterclockwise arc of the unit circle centered at v from s and the other with the clockwise arc of the unit circle centered at v from s', in $O(\log n)$ time, and get the first intersections u and u' of the arcs with ∂P. Then we check whether a vertex event occurs at the orientations of two unit chords vu and vu'. Thus we can find all vertex events in P in $O(n \log n)$ time.

Let O_T be the set of all top feasible orientations, and O_B be the set of all bottom feasible orientations. Note that O_T and O_B induce intervals contained in O and their endpoints are f-events by definition. Let $\mathcal{I}(O_T)$ and $\mathcal{I}(O_B)$ be the set of intervals induced by O_T and O_B, respectively. By Lemma 12, there are $O(n)$ intervals in $\mathcal{I}(O_T)$ and $\mathcal{I}(O_B)$. For a top feasible orientation $\theta \in O_T$, let $\bar{H}(\theta)$ be the largest unit histogon inscribed in P with top side lying on the top unit chord of orientation θ. Let $I \in \mathcal{I}(O_T)$ be an interval with endpoints θ_0, θ_1 satisfying $\theta_0 < \theta_1$. For any $\theta \in I$, the top side $\alpha(\theta) = \alpha_1(\theta)\alpha_2(\theta)$ of $\bar{H}(\theta)$ is a unit chord, and moves continuously as θ continuously increases in I by Lemma 6. Hence, its limits at the endpoints θ_0 and θ_1 are well defined as $\alpha(\theta_0)$ and $\alpha(\theta_1)$, so we can discuss the top side $\alpha(\theta)$ over all $\theta \in \mathsf{cl}(I)$ in the closure of each interval $I \in \mathcal{I}(O_T)$, including its endpoints.

It is possible that two consecutive intervals $I, I' \in \mathcal{I}(O_T)$ share an endpoint θ', while θ' belongs to none of I and I'. In this case, the limits of $\alpha(\theta)$ at θ' over I and I' may not agree. In the following, we handle each interval of $\mathcal{I}(O_T)$ separately in order, so we abuse a notation to mean $\alpha(\theta')$ by the limit over the interval we are currently handling. This will be clear from context.

4.1 Orientations dominated by other orientations

Let $\mathcal{C}(\theta)$ denote $\partial P[\alpha_1(\theta), \alpha_2(\theta)]$ for an orientation $\theta \in \mathsf{cl}(I)$ with $I \in \mathcal{I}(O_T)$. Since the interior angles at the endpoints of $\alpha(\theta)$ in $P[\alpha_2(\theta), \alpha_1(\theta)]$ are at least $\pi/2$, the endpoints of $\alpha(\theta)$ move continuously along ∂P in counterclockwise direction as θ continuously increases in I. This implies that $\mathcal{C}(\theta) \not\subseteq \mathcal{C}(\theta')$ and $\mathcal{C}(\theta') \not\subseteq \mathcal{C}(\theta)$ for any $\theta, \theta' \in \mathsf{cl}(I)$ with $\theta \neq \theta'$.

▶ **Observation 13.** *For $I \in \mathcal{I}(O_T)$ and $\theta \neq \theta' \in \mathsf{cl}(I)$, we have $\mathcal{C}(\theta) \not\subseteq \mathcal{C}(\theta')$ and $\mathcal{C}(\theta') \not\subseteq \mathcal{C}(\theta)$.*

▶ **Lemma 14.** *For $I, I' \in \mathcal{I}(O_T)$ with $I \neq I'$, $\theta \in \mathsf{cl}(I)$, and $\theta' \in \mathsf{cl}(I')$, it holds that $|\bar{H}(\theta)| < |\theta - \theta'|$ if $\mathcal{C}(\theta) \subseteq \mathcal{C}(\theta')$.*

For $I, I' \in \mathcal{I}(O_T)$ with $I \neq I'$, $\theta \in \mathsf{cl}(I)$, and $\theta' \in \mathsf{cl}(I')$, we say θ is *δ-dominated* by θ' (or θ' δ-dominates θ) if $\mathcal{C}(\theta) \subseteq \mathcal{C}(\theta')$ and $|\theta - \theta'| \leq \delta$. By Observation 13, θ and θ' are contained in two distinct intervals of $\mathcal{I}(O_T)$ if θ δ-dominates θ' or θ' δ-dominates θ. If θ is δ-dominated by θ', we have $|\bar{H}(\theta)| < \delta$ by Lemma 14. Observe, however, that there can be orientations $\theta'' \in O_T$ with $|\bar{H}(\theta'')| < \delta$ that are not dominated by any other orientations contained in an interval of $\mathcal{I}(O_T)$. We use the following lemma to determine whether an orientation $\theta \in I$ is δ-dominated by another orientation $\theta' \in I'$ for $I, I' \in \mathcal{I}(O_T)$.

▶ **Lemma 15.** *Let $\theta_1, \theta_2, \theta$ be three orientations, each contained in the closure of an interval in $\mathcal{I}(O_T)$, with $\theta_1 < \theta < \theta_2$ or $\theta_2 < \theta < \theta_1$. Suppose that $\mathcal{C}(\theta_1) \not\subseteq \mathcal{C}(\theta)$ and $\mathcal{C}(\theta) \not\subseteq \mathcal{C}(\theta_1)$. Then, we have $\mathcal{C}(\theta) \subseteq \mathcal{C}(\theta_2)$ if $\mathcal{C}(\theta_1) \subseteq \mathcal{C}(\theta_2)$, and $\mathcal{C}(\theta_2) \subseteq \mathcal{C}(\theta)$ if $\mathcal{C}(\theta_2) \subseteq \mathcal{C}(\theta_1)$.*

Lemma 15 provides us a tool to infer the δ-dominance relation over orientations in each interval of O_T, namely, if θ_1 is δ-dominated by θ_2, then θ is also δ-dominated by θ_2, and if θ_2 is δ-dominated by θ_1, then θ_2 is also δ-dominated by θ. We use this, together with Observation 13 and Lemma 14, to remove as much δ-dominated orientations as possible from each interval in $\mathcal{I}(O_T)$, resulting in relevant subintervals.

Removing δ-dominated orientations. Let $I, I' \in \mathcal{I}(O_T)$ be two distinct intervals with $\mathsf{cl}(I) = [\theta_0, \theta_1]$ and $\mathsf{cl}(I') = [\theta'_0, \theta'_1]$ such that an orientation $\theta \in \mathsf{cl}(I)$ is δ-dominated by an orientation $\theta' \in \mathsf{cl}(I')$. By Lemma 15, every orientation in $[\theta, \theta_1]$ is δ-dominated by θ'_0 if $\theta < \theta'$, and every orientation in $[\theta_0, \theta]$ is δ-dominated by θ'_1 if $\theta > \theta'$. Thus, we can determine whether an orientation is δ-dominated by another orientation using the endpoints of intervals in $\mathcal{I}(O_T)$. Moreover, the set of orientations in I which are not δ-dominated by other orientations appears as a subinterval of I unless it is empty.

In the following, we describe how to remove δ-dominated orientations from the intervals of $\mathcal{I}(O_T)$. Note that this procedure does not remove all δ-dominated orientations but the δ-dominated orientations that remain after the procedure have some property as shown in Lemma 16. We process the intervals of $\mathcal{I}(O_T)$ one by one in increasing order of orientation. In the course, we maintain a sequence L of (sub)intervals that have been processed so far after removing the orientations δ-dominated by some other orientations in the closure of an interval of $\mathcal{I}(O_T)$. Initially, L is set to an empty list.

Imagine that we have processed the first i intervals of $\mathcal{I}(O_T)$ and we are about to process the $(i+1)$-th interval I of $\mathcal{I}(O_T)$ with $\mathsf{cl}(I) = [\theta_0, \theta_1]$. Let $L = \langle [\theta_0^1, \theta_1^1], [\theta_0^2, \theta_1^2], \ldots, [\theta_0^m, \theta_1^m] \rangle$ be the sequence of intervals that have been processed so far. If L is empty, we simply append $\mathsf{cl}(I)$ into L. Otherwise, we update L by removing (sub)intervals of L or a (sub)interval of $\mathsf{cl}(I)$ consisting of the orientations δ-dominated by other orientations by the following rules. If $\theta_0 = \theta_1^m$, we set θ_1^m to be $\theta_1^m - \varepsilon$ for infinitesimally small $\varepsilon > 0$ temporarily whenever we check the δ-dominance between θ_0 and θ_1^m.

Rule 1. If θ_0 and θ_1^m do not δ-dominate each other, append $\mathsf{cl}(I) = [\theta_0, \theta_1]$ to L.
Rule 2. If θ_1^m is δ-dominated by θ_0,
- **a.** find the smallest integer $0 < j \leq m$ such that θ_1^j is δ-dominated by θ_0,
- **b.** find the smallest orientation $r \in [\theta_0^j, \theta_1^j]$ such that r is δ-dominated by θ_0,
- **c.** remove $[\theta_0^j, \theta_1^j], [\theta_0^{j+1}, \theta_1^{j+1}], \ldots, [\theta_0^m, \theta_1^m]$ from L, and
- **d.** append $[\theta_0^j, r]$ to L if $\theta_0^j < r$, and then append $\mathsf{cl}(I) = [\theta_0, \theta_1]$ to L.

Rule 3. If θ_0 is δ-dominated by θ_1^m,
- **a.** find the largest orientation $r \in [\theta_0, \theta_1]$ such that r is δ-dominated by θ_1^m, and
- **b.** append $[r, \theta_1]$ to L if $r < \theta_1$.

After removing δ-dominated orientations from the intervals of $\mathcal{I}(O_T)$ by the procedure above, the list L consists of $O(n)$ intervals and has the following property.

▶ **Lemma 16.** *For any two orientations θ, θ' contained in some intervals of L, $|\theta - \theta'| = \delta$ if θ is δ-dominated by θ'.*

Analysis. We maintain the list L of intervals in increasing order of orientation. Let $I \in \mathcal{I}(O_T)$ with $\mathsf{cl}(I) = [\theta_0, \theta_1]$ be the interval that we are about to process for $L = \langle [\theta_0^1, \theta_1^1], [\theta_0^2, \theta_1^2], \ldots, [\theta_0^m, \theta_1^m] \rangle$. The last interval $[\theta_0^m, \theta_1^m]$ of L can be found in $O(1)$ time. We can check if Rule 1 applies and append $[\theta_0, \theta_1]$ to L in $O(1)$ time. When Rule 2 applies, we find all intervals $[\theta_0', \theta_1']$ in L such that θ_1' is δ-dominated by θ_0 by checking the intervals in L one by one in decreasing order of orientation. By Lemma 15, those intervals form a contiguous subsequence $\langle [\theta_0^j, \theta_1^j], [\theta_0^{j+1}, \theta_1^{j+1}], \ldots, [\theta_0^m, \theta_1^m] \rangle$ of L. We remove them from L in time linear to the number of removed intervals. Then we find the smallest orientation r in $[\theta_0^j, \theta_1^j]$ such that r is δ-dominated by θ_0. If $\theta_0^j < r$, then either it holds that $|\theta_0 - r| = \delta$ or $\mathcal{C}(\theta_0)$ and $\mathcal{C}(r)$ share the common endpoint $\alpha_1(\theta_0) = \alpha_1(r)$. In the former case, we find $r \in [\theta_0^j, \theta_1^j]$ such that $|\theta_0 - r| = \delta$ in $O(1)$ time. In the latter case, we find $\alpha_2(r)$ using a circular arc query in $O(\log n)$ time with the directed arc of unit circle centered at $\alpha_1(\theta_0)$, since there is at most one orientation θ in $[\theta_0^j, \theta_1^j]$ such that $\alpha_1(\theta) = \alpha_1(\theta_0)$. When Rule 3 applies, we find the largest orientation $r \in [\theta_0, \theta_1]$ such that r is δ-dominated by θ_1^m for the last interval $[\theta_0^m, \theta_1^m]$ in L using a circular arc query in $O(\log n)$ time. Since there are $O(n)$ intervals in $\mathcal{I}(O_T)$, we can process them in $O(n \log n)$ time using $O(n)$ space.

4.2 Orientations θ with $|\bar{H}(\theta)| = \delta$.

Let I be an interval in L. Observe that the height of $\bar{H}(\theta)$ is positive and changes continuously as θ continuously increases in I since the endpoints of $\alpha(\theta)$ and the two chords orthogonal to $\alpha(\theta)$ through $\alpha_1(\theta)$ and $\alpha_2(\theta)$ move continuously along ∂P by Lemma 6. We call each orientation $\theta \in I$ such that $|\bar{H}(\theta)| = \delta$ a *δ-event orientation.* The set of δ-event orientations in I partitions I into subintervals I' such that either $|\bar{H}(\theta)| < \delta$ for all $\theta \in \mathsf{int}(I')$ or $|\bar{H}(\theta)| > \delta$ for all $\theta \in \mathsf{int}(I')$.

Finding δ-event orientations. We find all δ-event orientations contained in intervals of L as follows. A rectangle R of an orientation $\theta \in [0, \pi)$ is a copy of an axis-aligned rectangle $\bar{R}$ obtained by rotating $\bar{R}$ by θ in counterclockwise direction. The width of R is the width of $\bar{R}$, and the top and bottom sides of R are the ones corresponding to the top and bottom sides of $\bar{R}$, respectively.

First, we compute two lists L_T and L_B of intervals in addition to L. We compute the set O_T^δ of orientations θ such that the largest inscribed δ-width rectangle of orientation θ with top side lying on a chord of length δ has a positive height. Then we remove orientations $\theta \in I$ such that $|\theta - \theta'| \le 1/\delta$ and $\partial P[\gamma_1(\theta), \gamma_2(\theta)] \subseteq \partial P[\gamma_1(\theta'), \gamma_2(\theta')]$ for some other orientation $\theta' \in I'$ for $I, I' \in \mathcal{I}(O_T^\delta)$, where $\gamma_1(\theta)\gamma_2(\theta)$ is the top chord of length δ and orientation θ in P. This can be done by using the same procedure for computing L. Then we obtain L_T from the intervals of $\mathcal{I}(O_T^\delta)$. Similarly, we compute L_B from the set of orientations θ such that the largest inscribed δ-width rectangle of orientation θ with bottom side lying on a chord of length δ has a positive height.

If $|\bar{H}(\theta)| = \delta$ and $\bar{H}(\theta)$ has the bottom-left corner lying on ∂P for some $\theta \in O_T$, the largest inscribed δ-width rectangle of orientation $\theta + \pi/2$ with top side lying on a chord of length δ has height 1, and thus $\theta + \pi/2$ is contained in an interval of L_T. Similarly, $\theta - \pi/2$ is contained in an interval of L_B if $|\bar{H}(\theta)| = \delta$ and $\bar{H}(\theta)$ has the bottom-right corner lying on ∂P with height δ. Therefore, to compute δ-event orientations, we process the intervals of L, L_T and L_B in increasing order of orientation. For an interval $I \in L$, we can compute all v-event orientations in I corresponding to top unit chords and the subintervals into which I is partitioned by the v-event orientations. We can also compute the subintervals of L_T and

L_B using v-event orientations corresponding to top chords of length δ. Those subintervals are computed in increasing order of orientation and the combinatorial structure of the top chord (of length 1 or δ) of orientations θ is invariant for any θ contained in an interval.

Whenever a subinterval $I = [\theta_0, \theta_1]$ of L is identified by a v-event orientation, we check if there is an orientation $\theta \in I$ such that $|\bar{H}(\theta)| = \delta$. Let I' be the subinterval of L_T containing $\theta_0 + \pi/2$, among those partitioned by the v-event orientations for the top chords of length δ. Using the combinatorial structures of the top unit chord of orientation $\theta \in I$ and the top chord of length δ in I', we can build an equation that has a solution $\theta' \in I$ if and only if $|\bar{H}(\theta')| = \delta$. By solving the equation, we compute all orientations θ in I such that $|\bar{H}(\theta)| = \delta$ and $\bar{H}(\theta)$ has the bottom-left corner lying on ∂P. We also compute all δ-event orientations θ in I such that $\bar{H}(\theta)$ has the bottom-right corner lying on ∂P using $I = [\theta_0, \theta_1]$ and the subinterval of L_B containing $\theta_0 - \pi/2$. Similarly, whenever a subinterval of $L_{\pi/2}$ or $L_{-\pi/2}$ is identified by a v-event orientation, we compute all δ-event orientations in the subinterval. After processing all subintervals of L, L_T and L_B, we obtain all δ-event orientations.

δ-feasible subintervals. We partition the intervals of $\mathcal{I}(O_T)$ into subintervals by δ-event orientations. Then we can obtain the set of subintervals I' such that θ is a top feasible orientation and $|\bar{H}(\theta)| \geq \delta$ for all $\theta \in I'$. Similarly, we obtain the set of subintervals I'' from $\mathcal{I}(O_B)$ such that θ is a bottom feasible orientation and $|\bar{H}(\theta)| \geq \delta$ for all $\theta \in I''$. We apply the algorithms for computing the largest inscribed unit histogons in P of types C (Section 3.2) and D (Section 3.3) with the subintervals of $\mathcal{I}(O_T)$ and $\mathcal{I}(O_B)$ induced by δ-event orientations, instead of the intervals of $\mathcal{I}(O)$.

Analysis. Consider two orientations θ_1 and θ_2, each contained in an interval of L, such that both $\alpha_1(\theta_1)$ of $\alpha(\theta_1)$ and $\alpha_1(\theta_2)$ of $\alpha(\theta_2)$ lie on the same vertex of P. By Lemma 16, $|\theta_1 - \theta_2| \geq \delta$ since $\mathcal{C}(\theta_1) \subseteq \mathcal{C}(\theta_2)$ or $\mathcal{C}(\theta_2) \subseteq \mathcal{C}(\theta_1)$. Thus, for each vertex of P, there are $O(1/\delta)$ orientations in the intervals of L at which the top unit chord has its endpoint at the vertex. So there are $O(n/\delta)$ v-event orientations in the intervals of L. Similarly, we can show that the number of v-event orientations in the intervals of L_T and L_B is $O(\delta n) = O(n)$ since $\delta < 1$. Then, the total number of subintervals of L, L_T and L_B induced by v-event orientations is $O(n/\delta)$. Given L, L_T and L_B, we can compute the subintervals in $O(n/\delta)$ time while tracing the top chords (of length 1 or δ) in P.

Recall that the combinatorial structure of chords (of length 1 or δ) of orientations in the interior of a subinterval remains the same. Then the equation for computing δ-event orientations is of the form $\sin(\theta + A) + B = \delta \sin(\theta + C)$ using the law of sines, where A, B and C are all constants. Thus, the equation can be solved in $O(1)$ time, resulting in $O(1)$ solutions that correspond to δ-event orientations. Since there are $O(n/\delta)$ pairs of subintervals of L and L_T (and L and L_B) which overlap each other, there are $O(n/\delta)$ δ-event orientations in the intervals of $\mathcal{I}(O_T)$ and they can be computed in $O(n/\delta)$ time.

We compute L, L_T and L_B in $O(n \log n)$ time and the δ-event orientations in $O(n/\delta)$ time using $O(n)$ space. Moreover, the subintervals of $\mathcal{I}(O_T)$ induced by δ-event orientations have $O(n/\delta)$ v-event orientations in total. Thus, the running time of the algorithms for computing the largest histogon of types C and D in P decreases to $O(n \log n + n/\delta)$ if we use the subintervals of $\mathcal{I}(O_T)$ and $\mathcal{I}(O_B)$ induced by δ-event orientations, instead of the intervals of $\mathcal{I}(O)$.

▶ **Theorem 17.** *Suppose there is a unit histogon of height δ inscribed in P for $\delta < 1$. A largest inscribed unit histogon in P can be computed in $O(n \log n + n/\delta)$ time and $O(n)$ space.*

Our algorithm works under the assumption that there exists a unit histogon contained in P whose height is at least δ, while it can be used to test the existence for any $0 < \delta < 1$. Given δ as input, if there exists such a histogon, then Theorem 17 applies and our algorithm returns a largest inscribed unit histogon in P. Otherwise, if not, all orientations in O_T and O_B are removed by the δ-dominance relation or there is no δ-event orientation. Hence, this case can be identified when the list L is turned to be empty or when no δ-event orientation is identified. In the former case, it is obvious that there is no unit histogon of height at least δ. In the latter case, the height of the unit histogon $\bar{H}(\theta)$ is either larger than δ for all $\theta \in O_T$ or smaller than δ for all $\theta \in O_T$. Thus, by picking one orientation θ from O_T and computing $\bar{H}(\theta)$, one can check which case this is.

▶ **Corollary 18.** *Given $0 < \delta < 1$, one can determine whether there exists an inscribed unit histogon with height δ in P in $O(n \log n + n/\delta)$ time using $O(n)$ space.*

Note that we can compute a largest inscribed unit histogon in $O(n \log n)$ time if its height h is $\Omega(1/\log n)$ by setting $\delta = 1/\log n$ in Corollary 18 and Theorem 17. Otherwise, we can get an output-sensitive algorithm for finding a largest inscribed unit histogon as follows. We search a value h_0 with $0 < h_0 \leq h$ for the height h of a largest inscribed unit histogon using Corollary 18 with a sequence of δ values, $\delta_i = 2^{-2^i}$ for $i = 0, 1, \ldots, \lceil \log\log(1/h) \rceil$. Then we apply Theorem 17 with h_0. Observe that $2^{-2^m} \leq h < 2^{-2^{m-1}}$ for some nonnegative integer m. Then $m < \log\log(1/h) + 1$ and $2^{-2^m} > h^2$. This results in running time $O(n \log n \log\log(1/h) + n/h^2)$. We can even improve it by starting with the test value δ_j such that $\delta_j < 1/\log n \leq \delta_{j-1}$, concluding the following.

▶ **Corollary 19.** *Let h be the height of a largest inscribed unit histogon in P. A largest inscribed unit histogon in P can be computed in $O(n \log n)$ time if $h = \Omega(\frac{1}{\log n})$, or in $O(n \log n \log\log \frac{1}{h \log n} + \frac{n}{h^2})$ time, otherwise.*

5 Largest inscribed histogon and smallest circumscribed histogon

We also considered three optimization problems for a convex polygon P with n vertices. The term $T(m)$ appearing in the running times in the following denotes the time complexity of the optimization step for the trigonometric expression with $O(m)$ terms, each of quadratic form. Barequet and Rogol [4] observed that $T(m) = O(m)$ in practice.

- **Largest inscribed k-histogon.** Given a positive integer $k > 1$, find a largest k-histogon of arbitrary orientation inscribed in P. We present an algorithm for this problem that runs in $O(kn^2(\log n + kT(\min\{k, n\})))$ time using $O(\min\{k, n\}n)$ space.
- **Largest inscribed histogon.** Find the largest histogon of arbitrary orientation inscribed in P. We present an algorithm for this problem that runs in $O(Dn^2(\log n + T(\min\{D, n\})))$ time using $O(\min\{D, n\}n)$ space, where D denotes the diameter of P.
- **Smallest circumscribed histogon.** Find the smallest histogon of arbitrary orientation circumscribed to P. We present an algorithm for this problem that runs in $O(Dn(\log(\min\{D, n\}) + T(\min\{D, n\})))$ time using $O(n)$ space, where D denotes the diameter of P.

References

1 Alok Aggarwal, Maria M Klawe, Shlomo Moran, Peter Shor, and Robert Wilber. Geometric applications of a matrix-searching algorithm. *Algorithmica*, 2(1):195–208, 1987.

2 Hee-Kap Ahn, Sang Won Bae, Otfried Cheong, and Joachim Gudmundsson. Aperture-angle and hausdorff-approximation of convex figures. *Discrete & Computational Geometry*, 40(3):414–429, 2008.

3 Hee-Kap Ahn, Peter Brass, Otfried Cheong, Hyeon-Suk Na, Chan-Su Shin, and Antoine Vigneron. Inscribing an axially symmetric polygon and other approximation algorithms for planar convex sets. *Computational Geometry*, 33(3):152–164, 2006.

4 Gill Barequet and Vadim Rogol. Maximizing the area of an axially symmetric polygon inscribed in a simple polygon. *Computers & Graphics*, 31(1):127–136, 2007.

5 Binay Bhattacharya and Asish Mukhopadhyay. On the minimum perimeter triangle enclosing a convex polygon. In *Proc. Japanese Conference on Discrete and Computational Geometry (JCDCG 2002)*, pages 84–96, 2002.

6 Sergio Cabello, Otfried Cheong, Christian Knauer, and Lena Schlipf. Finding largest rectangles in convex polygons. *Computational Geometry*, 51:67–74, 2016.

7 Tauã M. Cabreira, Lisane B. Brisolara, and Paulo R. Ferreira Jr. Survey on coverage path planning with unmanned aerial vehicles. *Drones*, 3(1), 2019.

8 Jyun-Sheng Chang and Chee-Keng Yap. A polynomial solution for the potato-peeling problem. *Discrete & Computational Geometry*, 1(2):155–182, 1986.

9 Siu-Wing Cheng, Otfried Cheong, Hazel Everett, and Rene Van Oostrum. Hierarchical decompositions and circular ray shooting in simple polygons. *Discrete & Computational Geometry*, 32(3):401–415, 2004.

10 Yujin Choi, Seungjun Lee, and Hee-Kap Ahn. Maximum-area and maximum-perimeter rectangles in polygons. *Computational Geometry*, 94:101710, 2021.

11 Howie Choset. Coverage for robotics – A survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31:113–126, October 2001.

12 Jaehoon Chung, Sang Won Bae, Chan-Su Shin, Sang Duck Yoon, and Hee-Kap Ahn. Approximating convex polygons by histogons. In *Proc. 34th Canadian Conference on Computational Geometry (CCCG 2022)*, 2022.

13 Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars. *Computational Geometry: Algorithms and Applications*, chapter Point Location, pages 121–146. Springer-Verlag TELOS, 3rd edition, 2008.

14 A DePano, Yan Ke, and Joseph O'Rourke. Finding largest inscribed equilateral triangles and squares. In *Proc. 25th Allerton Conference on Communications, Control and Computing*, pages 869–878, 1987.

15 Enric Galceran and Marc Carreras. A survey on coverage path planning for robotics. *Robotics and Autonomous Systems*, 61(12):1258–1276, 2013.

16 Martin Held. *On the Computational Geometry of Pocket Machining.* Lecture Notes in Computer Science. Springer Verlag, 1991.

17 Kai Jin and Kevin Matulef. Finding the maximum area parallelogram in a convex polygon. In *Proc. 23rd Canadian Conference on Computational Geometry (CCCG 2011)*, 2011.

18 Christian Knauer, Lena Schlipf, Jens M Schmidt, and Hans Raj Tiwary. Largest inscribed rectangles in convex polygons. *Journal of discrete algorithms*, 13:78–85, 2012.

19 Seungjun Lee, Taekang Eom, and Hee-Kap Ahn. Largest triangles in a polygon. *Computational Geometry*, 98:101792, 2021.

20 Elefterios A. Melissaratos and Diane L. Souvaine. On solving geometric optimization problems using shortest paths. In *Proc. 6th Annual Symposium on Computational Geometry (SCG '90)*, pages 350–359, 1990.

21 Joseph O'Rourke, Alok Aggarwal, Sanjeev Maddila, and Michael Baldwin. An optimal algorithm for finding minimal enclosing triangles. *Journal of Algorithms*, 7(2):258–269, 1986.

22 Christian Schwarz, Jürgen Teich, Alek Vainshtein, Emo Welzl, and Brian L. Evans. Minimal enclosing parallelogram with application. In *Proc. 11th Annual Symposium on Computational Geometry (SCG '95)*, pages 434–435, 1995.

23 Godfried Toussaint. Solving geometric problems with the rotating calipers. In *Proc. Mediterranean Electrotechnical Conference (MELECON'83)*, 1983.

24 Ivor van der Hoog, Vahideh Keikha, Maarten Löffler, Ali Mohades, and Jérôme Urhausen. Maximum-area triangle in a convex polygon, revisited. *Information Processing Letters*, 161:105943, 2020.