Available online at www.sciencedirect.com

ScienceDirect

Computational Geometry 38 (2007) 154–169

On finding widest empty curved corridors

Sergey Bereg [a], J. Miguel Díaz-Báñez [b,*,1], Carlos Seara [c,2], Inmaculada Ventura [d,1]

[a] Department of Computer Science, University of Texas at Dallas, Box 830688, Richardson, TX 75083, USA
[b] Departamento de Matemática Aplicada II, Universidad de Sevilla, Camino de los Descubrimientos s/n, 41092 Sevilla, Spain
[c] Departament de Matemàtica Aplicada II, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain
[d] Departamento de Matemáticas, Universidad de Huelva, Spain

Received 21 December 2004; received in revised form 13 December 2006; accepted 21 February 2007

Available online 3 March 2007

Communicated by G. Toussaint

Abstract

An α-siphon of width w is the locus of points in the plane that are at the same distance w from a 1-corner polygonal chain C such that α is the interior angle of C. Given a set P of n points in the plane and a fixed angle α, we want to compute the widest empty α-siphon that splits P into two non-empty sets. We present an efficient $O(n \log^3 n)$-time algorithm for computing the widest oriented α-siphon through P such that the orientation of a half-line of C is known. We also propose an $O(n^3 \log^2 n)$-time algorithm for the widest arbitrarily-oriented version and an $\Theta(n \log n)$-time algorithm for the widest arbitrarily-oriented α-siphon anchored at a given point.

Keywords: Corridors; Geometric optimization; Facility location

1. Introduction

Let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of n points in the plane. A corridor through P is the open region of the plane bounded by two parallel lines intersecting the convex hull of P, $CH(P)$. A corridor is empty if it contains no points of P. Houle and Maciel [12] solved the problem of computing the widest empty corridor through P in $O(n^2)$ time (Fig. 1(a)). One motivation of the widest empty corridor problem is to find a collision-free route for transport objects through a set of obstacles (points). However, even the widest empty corridor may be not wide enough. This motivates to allow angle turns. Cheng [1,2] studied this generalization considering an L-*shaped corridor*, which is the concatenation of two perpendicular links (a link is composed by two parallel rays and one line segment forming an unbounded trapezoid). Díaz-Báñez and Hurtado [4] proposed an $O(n^2)$-time algorithm for locating an obnoxious 1-corner polygonal chain anchored at two given points.

* Corresponding author.
E-mail addresses: besp@utdallas.edu (S. Bereg), dbanez@us.es (J.M. Díaz-Báñez), carlos.seara@upc.edu (C. Seara), iventura@us.es (I. Ventura).
[1] Partially supported by grants BFM2003-04062 and MTM2006-03909.
[2] Partially supported by projects MEC MTM2006-01267 and DURSI 2005SGR00692.

0925-7721/$ – see front matter
doi:10.1016/j.comgeo.2007.02.003

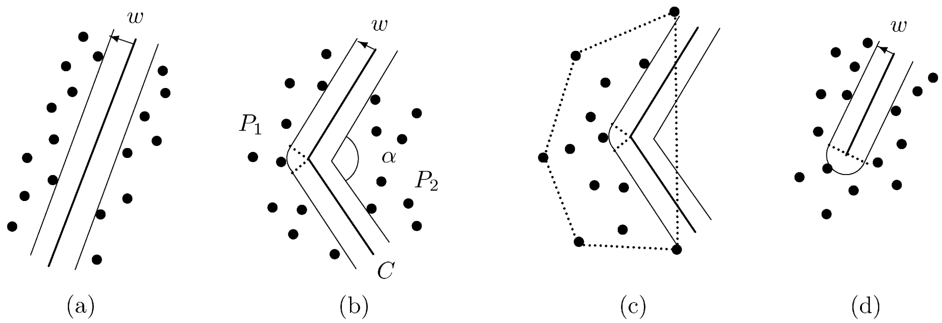

Fig. 1. (a) Corridor, (b) α-siphon, (c) unbounded width siphon, (d) silo.

In this paper we study a kind of corridor which we call a *siphon* that we define next. Let C be a 1-corner polygonal chain consisting of two half-lines emanating from a point v, the *corner* of C. Let α be the *interior angle* of C. By $d(p,q)$ we denote the Euclidean distance between two points $p, q \in \mathbb{R}^2$. The distance between p and C is defined as

$$d(p, C) = \min\{d(p,q) \colon q \in C\}.$$

Definition 1. A *siphon* is the locus of points in the plane that are at distance w from C, where w is called the *siphon width*. An α-siphon is a siphon such that the interior angle α of C, the *siphon angle*, is fixed.

An α-siphon is determined by C and w; if $\alpha \neq \pi$, it is bounded by an exterior boundary (formed by a circular arc joining two half-lines) and an interior boundary (formed by two half-lines).

The widest α-siphon problem. Given a set P of n points in the plane and a fixed value α, $0 \leqslant \alpha \leqslant \pi$, compute the α-siphon with the largest width w such that no points of P lies in its interior, and intersects $CH(P)$ producing a non-trivial bipartition P_1, P_2, of P (Fig. 1(b)).

The condition of producing a partition $P_1 \neq \emptyset$, $P_2 \neq \emptyset$ of P avoids to obtain an α-siphon "scratching the exterior" of P without actually passing through P and, therefore, being arbitrarily wide (Fig. 1(c)). For any 1-corner polygonal chain C which contains no points of P and produces a non-trivial partition of P, there always exists an α-siphon defined by C. In the sequel, unless otherwise is specified, we assume that the α-siphon is empty of points.

Notice that a π-siphon is just a corridor [12] where C is a line (Fig. 1(a)). Follert et al. [9] defined a *silo* as the locus of points in the plane that are at distance w from a ray (Fig. 1(d)), they partially studied the problem of computing the widest empty silo for P giving an optimal $O(n \log n)$-time algorithm in the case that the endpoint of ray is anchored at a given point. As far as we know, the non-anchored case is still unsolved. A 0-siphon is not a silo because the non-trivial partition condition. Follert et al. [9] also solved the corridor problem in $O(n \log n)$ time when the axis of the corridor pass through a given point.

The widest α-siphon gives a "better" solution than the L-shaped corridor of Cheng in the following sense: Suppose that we are interesting in transporting a circular object (a disk) with radius w through a set P of obstacles (points), and we ask for the decision problem: *Is there a* 1*-corner polygonal path for transporting the disk through P without collision*? Cheng's algorithm can produces a negative answer while our algorithm gives an affirmative answer; this is so because the width of the widest α-siphon is always larger than or equal to the width of the widest L-shaped corridor. In fact, every L-shaped corridor contains a siphon of the same width and the reciprocal is false (Fig. 2). Notice that an α-siphon is the area swept by a disk whose center describes the 1-corner polygonal chain C. The siphon shape has been also considered by Glozman et al. in [10] where they studied the problem of covering points with a siphon of smallest width.

The widest α-siphon problem belongs to the geometric optimization area in which the placement of a particular kind of empty geometric object of "maximum measure" is considered [3,5,7,16,17,21]. Some possible applications of the widest α-siphon problem fall into the facility location area, where the usual goal is to locate an object (*facility*) within an underlying space such that some distance between the facility and the given points (*sites*) is minimized or maximized; see [8] for a recent survey on the current state-of-art. Some of the problems are concerned with finding

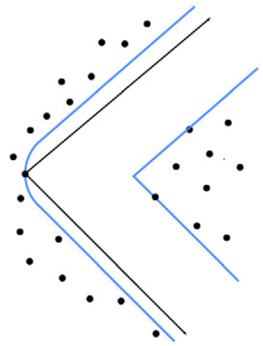

Fig. 2. The width of the widest α-siphon is larger than the width of the widest L-corridor.

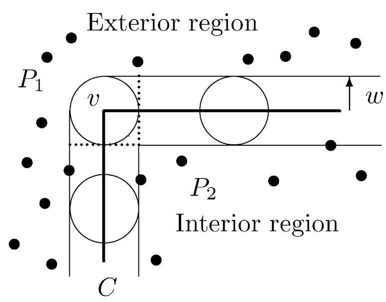

Fig. 3. A S–E orthogonal siphon of width w.

an "undesirable" or "obnoxious" structure, just maximizing the smallest distance between the facility and the given points. In this sense, our problem asks for the optimal placement of an obnoxious 1-corner polygonal chain with interior angle α.

In this paper three variants of the widest α-siphon problem are considered: (i) the *widest oriented α-siphon problem*, where we know the angle α and the orientation of one of the half-lines of C; (ii) the *widest arbitrarily-oriented α-siphon problem*, where only the angle α is known; and (iii) the *widest anchored and arbitrarily-oriented α-siphon problem*, where the corner of C is anchored at a given point. In Section 2 we present an efficient $O(n \log^3 n)$-time algorithm for the first problem. We study the arbitrarily-oriented case showing an $O(n^3 \log^2 n)$-time algorithm for this problem in Section 3. Finally, in Section 4 we solve the problem of computing the widest anchored and arbitrarily-oriented siphon with an optimal $\Theta(n \log n)$-time algorithm.

2. Widest oriented α-siphon

We study the problem of computing the widest oriented α-siphon for P. First of all, we consider the problem of computing the widest orthogonal siphon to be defined next, and then we will solve the general case.

An *orthogonal siphon* is an oriented $\frac{\pi}{2}$-siphon such that the half-lines of C are vertical and horizontal (Fig. 3). There are four possible orthogonal siphons according to the north, south, east, and west orientations of the half-lines of C: S–E, N–E, N–W, and S–W. We only consider the S–E case, other cases can handled analogously. Next, we show how to solve the decision problem of the S–E orthogonal siphon for P.

Orthogonal siphon decision problem. Given P and a real value $w > 0$, determine whether there exists a S–E orthogonal siphon of width w.

A S–E orthogonal siphon of width w is specified by the corner $v = (x_v, y_v)$ of C (Fig. 3). The corner and C have the following properties:

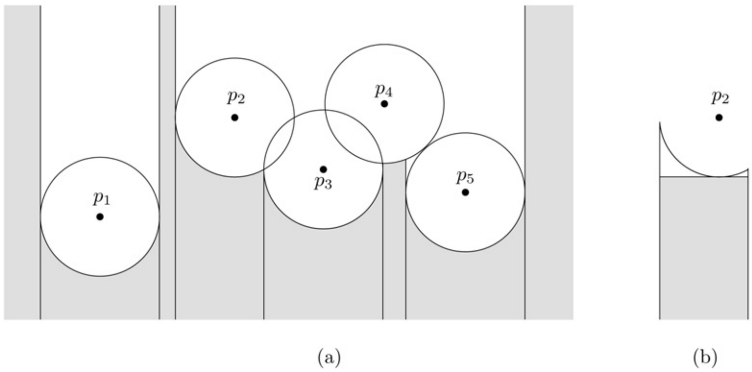

Fig. 4. (a) The region $V(P, w)$, (b) support of the region $R_V(p_2)$.

1. *Vertical property*. For any point $u \in \{(x_v, y) \mid y \leqslant y_v\}$, the interior of the disk centered at u and radius w contains no points of P.
2. *Horizontal property*. For any point $u \in \{(x, y_v) \mid x \geqslant x_v\}$, the interior of the disk centered at u and radius w contains no points of P.
3. *Separation property*. Both the exterior and interior regions of C contain at least a point of P.

We define $V(P, w)$ as the locus of points in the plane satisfying the vertical property. Similarly, $H(P, w)$ is the locus of points in the plane satisfying the horizontal property.

Lemma 1. *$V(P, w)$ and $H(P, w)$ can be constructed in* $O(n \log n)$ *time.*

Proof. We only show how to construct $V(P, w)$, the construction of $H(P, w)$ is similar. Place n disks of radius w at all the points p_i of P. Then $V(P, w)$ is the set of points visible from $y = -\infty$ (Fig. 4). It consists of at most $2n + 1$ regions where each region is either a vertical slab or a set $R_V(p_i)$ of points below an arc corresponding to a point p_i.

Let $L_V(P, w)$ be the list of these regions. $V(P, w)$ can be easily computed. Sort P by y-coordinate. Let $P' = \emptyset$. Insert points into P' in the decreasing order of its y-coordinates and maintain $L_V(P', w)$. When a point p_i is inserted, locate x_{p_i} in the regions $L_V(P', w)$ and update $L_V(P', w)$ by checking the neighbor regions. An useful property of $V(P, w)$ is that every point p_i has at most one arc in $V(P, w)$, since the disks have the same radius.

Updating $L_V(P', w)$. Let $R_1, R_2, \ldots$ be the regions in $L_V(P', w)$. Let $[x_j, x_{j+1}]$ be the projection of a region R_j onto the x-axis. Suppose that $x_{p_i} \in [x_j, x_{j+1}]$. Let D_i be the disk with center at p_i and radius w. Suppose that the contribution of D_i is an arc a_i whose projection onto x-axis is in the interval $[x_k, x_l]$, where k is the largest index and l is the smallest index. Then the algorithm checks $l - k + 1$ regions $R_k, R_{k+1}, \ldots, R_l$ and $l - k - 1$ regions $R_{k+1}, \ldots, R_{l-1}$ are deleted. At least half of the deleted regions correspond to disks (not vertical slabs) and we can charge them to the update step. The total time for all updates is $O(n)$ since every disk can be deleted only once. Notice that an arc of a point p_i can be modified many times. □

Let p_i be a point of P such that $V(P, w)$ contains an arc corresponding to p_i. We define a *V-support* of p_i, denoted by $S_V(p_i)$, as the largest rectangular region in $R_V(p_i)$ (Fig. 4(b)). Similarly, we define a *H-support* of p_i, $S_H(p_i)$.

Lemma 2. *Given P and w, there exists a S–E orthogonal siphon of width w if and only if one of the following conditions holds*:

(i) *there are two intersecting supports $S_V(p_i)$ and $S_H(p_j)$ such that a 1-corner polygonal chain with the corner at $S_V(p_i) \cap S_H(p_j)$ splits P, or*
(ii) *there is a point p_i of P whose arcs in $V(P, w)$ and $H(P, w)$ intersect, and the 1-corner polygonal chain with the corner at a point p in the intersection of these two arcs splits P.*

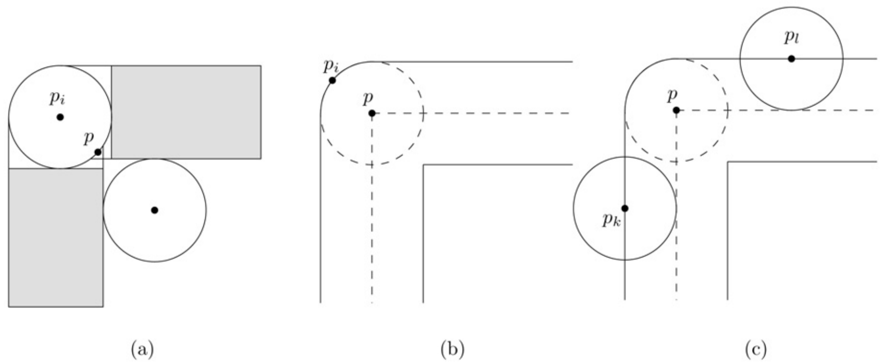

Fig. 5. (a) The point p is a common vertex of two arcs of p_i in $V(P,w)$ and $H(P,w)$. Notice that $S_V(p_i)$ and $S_H(p_i)$ (shaded rectangles) do not intersect. (b) and (c) two cases of an orthogonal siphon.

Proof. First, we show that if (i) or (ii) holds, then there is a S–E orthogonal siphon. If (i) holds, then the S–E orthogonal siphon is defined by a 1-corner polygonal chain with the corner at $S_V(p_i) \cap S_H(p_j)$. If (ii) holds, then the S–E orthogonal siphon is defined by a 1-corner polygonal chain with the corner at any common point p of the two arcs of p_i (Fig. 5(a)).

Conversely, suppose that there exists a S–E orthogonal siphon. Let p be the center of its arc. By translating the siphon up and to the left we can assume one of the following cases.

Case 1. The arc of the siphon contains a point $p_i \in P$ (Fig. 5(b)). Then p is in the intersection of the arcs of p_i in $V(P,w)$ and $H(P,w)$.

Case 2. The left boundary of the siphon contains a point p_k and the top boundary of the siphon contains a point p_l (Fig. 5(c)). Then p lies in a vertical slab or in the *V-support* of a point (the point must have y-coordinate at least y_{p_k}). Similarly p lies in a horizontal slab or in the *H-support* of a point (the point must have x-coordinate at most x_{p_k}). Therefore p satisfies (i). □

In [18] the maxima problem for points in the plane is considered. Concretely, given two points p_i and p_j, the following dominance relation is established:

Definition 2. (See [18].) The point p_i dominates the point p_j ($p_j \prec p_i$), if $x_{p_j} \leqslant x_{p_i}$ and $y_{p_j} \leqslant y_{p_i}$.

The relation $\prec$ is a partial order in P. A point $p_i \in P$ is called *maximal* if there does not exists $p_j \in P$ such that $i \neq j$ and $p_i \prec p_j$. The maxima problem consists of finding all the maximal points of P under dominance. The next theorem solves the maxima problem.

Theorem 1. *(See [13].) The maxima problem for P can be solved in $\mathrm{O}(n \log n)$ time. In the comparison-tree model, any algorithm that solves the maxima problem in the plane requires $\Omega(n \log n)$ time.*

We consider four maxima problems, one for each quadrant in the plane. Each of these problems is obtained by an assignment of signs $+$ or $-$ to each of the coordinates of the points of P. The formulation above corresponds to the assignment $++$. The set of maximal points for this assignment forms a *first quadrant staircase*. In our S–E orthogonal siphon problem we are interested in the maxima problem for the four quadrant, i.e., the dominance relation is defined as follows:

$$p_j \prec p_i \iff x_{p_j} \leqslant x_{p_i} \text{ and } y_{p_j} \geqslant y_{p_i}.$$

The corresponding maximal points form a *four quadrant staircase* (Fig. 6). Let $P_{\max}$ be the set of these maximal points for P. By Theorem 1, $P_{\max}$ can be computed in $\Theta(n \log n)$ time.

Remember that condition (ii) of Lemma 2 says that “the 1-corner polygonal chain splits P”. Next we show that condition (ii) can be replaced by the following equivalent condition:

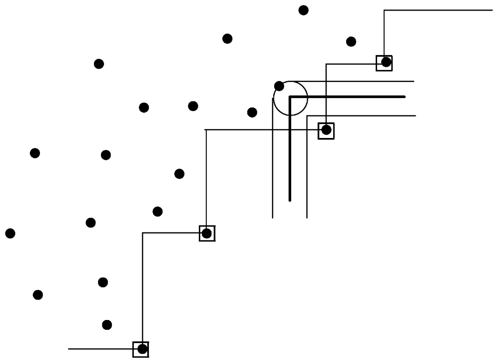

Fig. 6. The squared points are the maximal points for P.

(ii) there is a point $p_i \in P_0 = P \setminus P_{\max}$ whose arcs in $V(P, w)$ and $H(P, w)$ intersect.

The equivalence follows from the fact that the 1-corner polygonal chain with corner at a point p in the intersection of the arcs in $V(P, w)$ and $H(P, w)$ splits P, because $p_i \notin P_{\max}$ (Fig. 6).

Theorem 2. *The decision problem for the S–E orthogonal siphon can be solved in* $O(n \log n)$ *time.*

Proof. We construct $V(P, w)$ and $H(P, w)$ as described in Lemma 1. For every support rectangle R in $H(P, w)$ (either a horizontal slab or the H-support of a point), we do the following. (Notice that, in the case (i) of Lemma 1, we can assume that there is a point of P above the horizontal slab containing R, see Fig. 5(c)). We find the leftmost rectangle in $V(P, w)$ intersecting R, if any. Then we check if these two rectangles define a 1-corner polygonal chain with at least one point of P in its interior. Thus, by Lemma 2, the 1-corner polygonal chain defines a S–E orthogonal siphon.

If a S–E orthogonal siphon is not found in the previous steps, then we check the equivalent condition (ii) of Lemma 2 for all points $p_i \in P_0$.

Running time. The rectangles are sorted and standard data structures can be used to check the condition (i) of Lemma 2 in $O(n \log n)$ time. The condition (ii) of Lemma 2 takes $O(n)$ time. □

Theorem 3. *An optimal S–E orthogonal siphon through* P *can be found in* $O(n \log^3 n)$ *time.*

Proof. We apply the parametric searching technique [15] to compute the optimal siphon of width w^*. We need a parallel algorithm for solving the decision problem. Our parallel algorithm constructs $V(P, w)$ and $H(P, w)$, and then does the steps of Theorem 2 in parallel. We show how the set $V(P, w)$ can be computed using a parallel algorithm. We use the divide-and-conquer approach where the set P is divided into two sets P_1 and P_2 of size at most $\lceil n/2 \rceil$. The "conquer" step is as follows.

First, we intersect the intervals of $V(P_1, w)$ and $V(P_2, w)$. We assign one processor to every interval I. The intersection of the regions of $V(P_1, w)$ and $V(P_2, w)$ corresponding to I can be done in $O(1)$ time. Thus, the conquer step takes $O(1)$ time and the total time is $O(\log n)$. The running time of the optimization algorithm is

$$O\big(\big(P(n) + T_s(n)\big) T_p(n) \log P(n)\big),$$

where $T_s(n)$ is the running time of the sequential algorithm, $T_p(n)$ is the running time of the parallel algorithm, and $P(n)$ is the number of processors. The theorem follows since, by Theorem 2, $T_p(n) = O(\log n)$, and $P(n) = O(n)$. □

Remarks.

1. The above technique can be adapted for computing the widest oriented α-siphon defined by a 1-corner polygonal chain with a fixed direction of one of its half-lines and a given angle α, $0 < \alpha < \pi$. For the decision problem we have to change the horizontal property by the property corresponding to the orientation given by the siphon angle

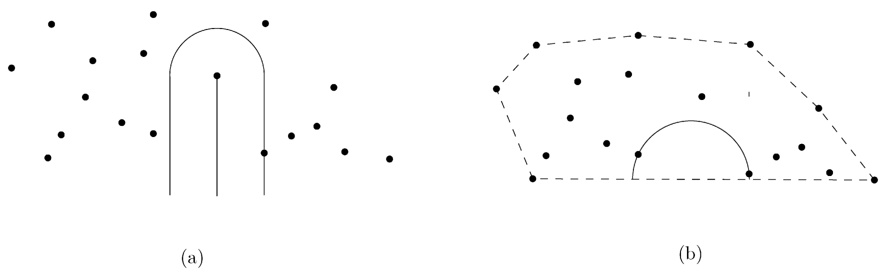

Fig. 7. The oriented 0-siphon problem.

α with respect to the vertical. The maxima problem can be adapted using a coordinate system which axes form an angle α. The rest of the algorithm is straightforward.

2. For $\alpha = \pi$, the oriented α-siphon is an oriented corridor. Assume that the orientation of the corridor is vertical. Thus, we can project the points of P on a horizontal line, sort the projected points, and compute the widest interval defined by two consecutive projected points, which is the width of the widest oriented corridor. The total time is $O(n \log n)$.
3. For $\alpha = 0$, we assume that the orientation of the 0-siphon is the vertical down and it produces a non-trivial bipartition of P. We suppose that the endpoint of its half-line is inside $CH(P)$. Then the oriented 0-siphon problem is solved in $O(n \log n)$ time as follows:
   Sweep P by a vertical line moving up and for every point $p \in P$ compute its two neighbors in the sorted order of their x-coordinates. Take the closest x-coordinate q_x. We store distance $d_x(p) = |p_x - q_x|$ for the point p. Next, we compute the distance $d(p)$ from every point p to its nearest neighbor using the Voronoi diagram. The width of the largest 0-siphon at p is $\min\{d_x(p), d(p)\}$. We compute the maximum over all $p \in P$ (Figs. 7(a) and (b)).

By the remarks above we have the following theorem.

Theorem 4. *The widest oriented α-siphon through P, $0 \leqslant \alpha \leqslant \pi$, can be computed in* $O(n \log^3 n)$ *time and* $O(n)$ *space.*

Notice that the widest oriented α-siphon through P may be not unique, several solutions, say k, of the same width are possible. Once we have computed the widest one, we can modify the decision algorithm and run it one more time. Thus, the algorithm can find all k solutions in $O(k + n \log n)$ time. Notice that k can be quadratic. A preliminary version of this problem appears in [6] with an $O(n^2)$ time algorithm which compute all the optimal solutions.

A similar parametric searching approach (by using squares instead of circles) can be used to compute the widest L-shaped orthogonal corridor of [1,2] with the same running time. For the L-shaped orthogonal corridor we do not consider condition (ii) in Lemma 2. Moreover, the technique can be applied for computing a corridor of the same kind but with an angle different from $\frac{\pi}{2}$ and knowing the orientation of one of its links. The complexity obtained by Cheng [1] for this particular case is $O(n^2 \log^5 n)$. We have the following result,

Theorem 5. *The widest orthogonal L-shaped corridor can be computed in* $O(n \log^3 n)$ *time and* $O(n)$ *space.*

2.1. Lower bound

Next we show a lower bound for the problem of computing the widest oriented α-siphon.

Theorem 6. *The problem of computing the widest oriented α-siphon has an* $\Omega(n \log n)$ *time lower bound in the algebraic decision tree model.*

Proof. We use a reduction to the MAX-GAP problem for points on a line. Let $x_1, \ldots, x_n$ be an instance of the MAX-GAP problem, i.e., a set of n real numbers. We proceed as follows.

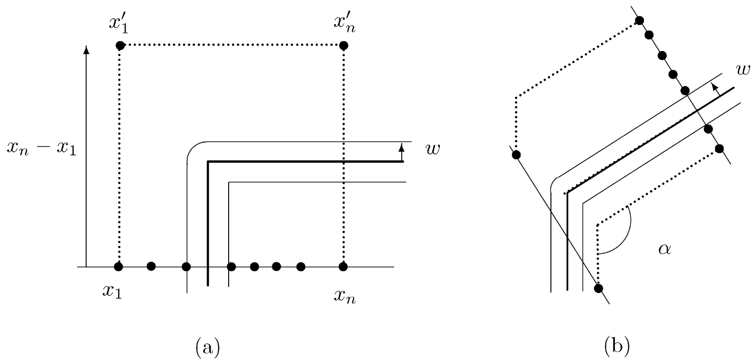

Fig. 8. Lower bound for the oriented α-siphon problem.

Compute $\max\{x_1, \dots, x_n\}$ and $\min\{x_1, \dots, x_n\}$ in linear time. Without loss of generality, assume that $x_1 = \min\{x_1, \dots, x_n\}$ and $x_n = \max\{x_1, \dots, x_n\}$. Put two points x'_1 and x'_n on vertical lines passing through x_1 and x_n, respectively, such that $x'_1 - x_1 = x'_n - x_n = x_n - x_1$ as it is illustrated in Fig. 8(a). Now we compute the widest orthogonal S–E siphon through the set of points $\{x_1, \dots, x_n, x'_1, x'_n\}$. Let w be the width of this widest orthogonal siphon. By construction, the width of the widest orthogonal siphon does not depend on the points x'_1, x'_n. Then the maximum gap for $x_1, \dots, x_n$ is $2w$.

For the widest oriented α-siphon problem with fixed α, the construction above can be easy modified as in Fig. 8(b). As the MAX-GAP problem has an $\Omega(n \log n)$ time lower bound in the algebraic decision tree model [14], the theorem follows. □

Still there is a gap between the $O(n \log^3 n)$ upper bound and the $\Omega(n \log n)$ lower bound. Notice that the proof of Theorem 6 also shows that the widest orthogonal L-shaped corridor problem has an $\Omega(n \log n)$ time lower bound in the algebraic decision tree model.

3. The widest arbitrarily-oriented α-siphon

In this section we deal with the computation of the widest arbitrarily-oriented α-siphon for P, that is, the orientation of the α-siphon is arbitrary and we only fix the *siphon angle* α. We assume that α is $\frac{\pi}{2}$; other values of α, $0 < \alpha < \pi$ can be handled analogously.

Lemma 3. *There always exists an optimal $\frac{\pi}{2}$-siphon for P such that its interior boundary contains either two points of P (one per each half-line) or only one point if this point is on the corner of the interior boundary.*

Proof. Otherwise, we can *move* the $\frac{\pi}{2}$-siphon till we get two points in the interior boundary. □

The points of P which determine a tentative placement of an optimal $\frac{\pi}{2}$-siphon are called *critical points*. We classify cases for critical points according to their location on the boundary of the $\frac{\pi}{2}$-siphon, obtaining the six cases (up to symmetry) illustrated in Fig. 9.

At most four points are necessary to construct an optimal $\frac{\pi}{2}$-siphon (three points for case (6)). An exhaustive, but straightforward, analysis of the situations that may arise gives the six cases. We omit here their descriptions, but the idea is to use the freedom left in order to move the $\frac{\pi}{2}$-siphon increasing its width until at least one point of P is encountered. With this characterization, we can exhaustively consider all possible cases and find the optimal $\frac{\pi}{2}$-siphon in $O(n^5)$ time. Next we show how to solve the problem more efficiently using Lemma 3.

The general process is to compute the widest arbitrarily-oriented $\frac{\pi}{2}$-siphon supported by a fixed pair of points (p_i, p_j) of P in its interior boundary in $O(n \log^2 n)$ time and $O(n \log n)$ space, and then repeat this computation for all the $O(n^2)$ pairs of points in total $O(n^3 \log^2 n)$ time.

First, we introduce some notation. Let (p_i, p_j) be a fixed pair of points of P and suppose that $x_{p_i} < x_{p_j}$. Assume that the coordinate system has the origin at p_i, and the segment $\overline{p_i p_j}$ is on the OX^+ axis. We will consider the

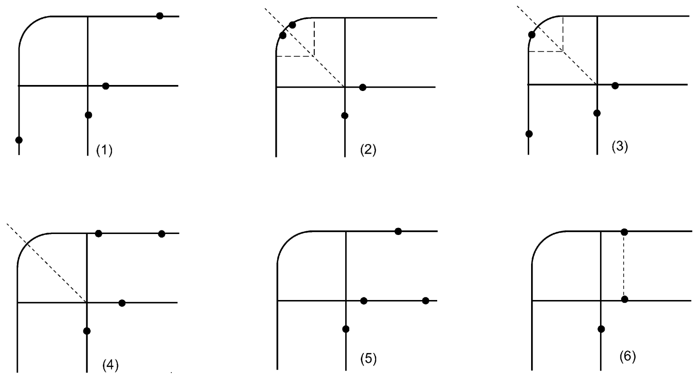

Fig. 9. Types of candidate siphons.

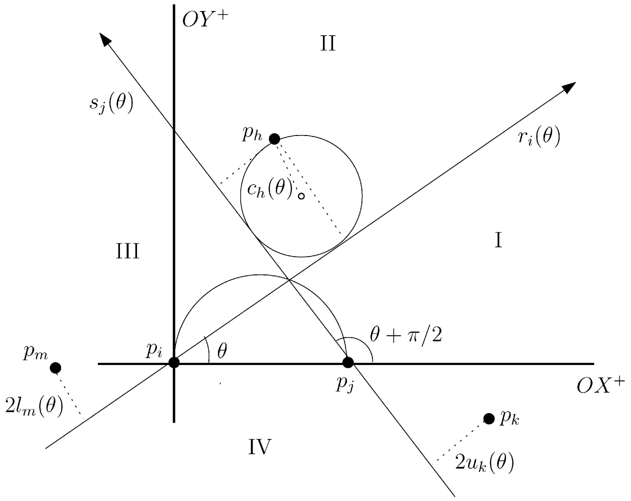

Fig. 10. Supported lines $r_i(\theta)$ and $s_j(\theta)$.

counterclockwise rotation of the two (perpendicular) supported lines $r_i(\theta)$ (oriented line passing through p_i with slope θ) and $s_j(\theta)$ (oriented line passing through p_j with slope $\theta + \frac{\pi}{2}$), to obtain the widest empty $\frac{\pi}{2}$-siphon supported by p_i and p_j. Thus, we can define the *orientation* of the $\frac{\pi}{2}$-siphon by the angle θ (Fig. 10).

We start the rotation of the oriented lines $r_i(\theta)$ and $s_j(\theta)$ at $\theta = 0$. The intersection point of $r_i(\theta)$ and $s_j(\theta)$ describes a half-circle with diameter $\overline{p_i p_j}$ as $\theta \in [0, \frac{\pi}{2}]$ (Fig. 10). The other half-circle will be considered for the pair (p_j, p_i). If $\alpha \neq \frac{\pi}{2}$ the intersection point describes an arc of circle defined by the chord $\overline{p_i p_j}$ and the angle α.

The directed lines $r_i(\theta)$ and $s_j(\theta)$ partition P into four disjoint subsets labeled as I, II, III, and IV, corresponding to the four quadrants (Fig. 10). Changing the orientation continuously, the partition survives till some of the two lines bumps a new point $p_k \in P \setminus \{p_i, p_j\}$.

For each point p_k in the regions I, II or III and a fixed angle, we can built a tentative $\frac{\pi}{2}$-siphon having p_i, p_j and p_k on their boundary. Our approach requires the study of the behavior of the width of a rotating $\frac{\pi}{2}$-siphon supported by p_i and p_j. Given a point $p_k \in P \setminus \{p_i, p_j\}$, we can generate all siphon-shaped corridors for whose p_i, p_j and p_k are on the boundary for $\theta \in [0, \frac{\pi}{2}]$. The definition of the corresponding width depends on the location of p_k according with regions I, II and III. Notice that when p_k lies on the region IV no $\frac{\pi}{2}$-siphon is generated by the pair (p_i, p_j). We define the width-functions as follows:

If $p_k \in$ I, let $u_k(\theta) = \frac{1}{2} d(p_k, s_j(\theta + \frac{\pi}{2}))$. If $p_k \in$ III, let $l_k(\theta) = \frac{1}{2} d(p_k, r_i(\theta))$. For a point p_k in subset II, let $C_k(\theta)$ be the (minimum radius) circle passing through p_k and tangent both to $r_i(\theta)$ and $s_j(\theta)$. Denote by $c_{ijk}(\theta)$ the center of $C_k(\theta)$. Thus, we define $c_k(\theta) = d(p_k, c_{ijk}(\theta))$ and $u_k(\theta)$, $l_k(\theta)$ as before. In this case, the width of the $\frac{\pi}{2}$-siphon is determined by $\min\{u_k(\theta), l_k(\theta), c_k(\theta)\}$.

The computation of the circle $C_k(\theta)$ is the well known Apolonio's problem [11]. For a fixed orientation θ, the values $u_k(\theta)$, $l_k(\theta)$ and $c_k(\theta)$ can be computed in constant time. Moreover, for each point p_k we can compute in O(1) time the range of each function $u_k(\theta)$, $l_k(\theta)$ and $c_k(\theta)$. Thus, in $O(n)$ time we generate all width-functions for all points $p_k \in P \setminus \{p_i, p_j\}$.

Let $\mathcal{A}$ be the arrangement of monotone Jordan arcs $u_k(\theta)$, $l_k(\theta)$ and $c_k(\theta)$, $\theta \in [0, \frac{\pi}{2}]$, for all the points $p_k \in P \setminus \{p_i, p_j\}$. The calculation of the widest empty siphon for a pair (p_i, p_j) can be reduced to compute the highest vertex of the lower envelope of $\mathcal{A}$. Although u_k and l_k are unimodal trigonometric functions, the geometric study of c_k is complicated. The following results establish a bound on the number of intersections between two curves of the arrangement by using their analytical expressions.

Lemma 4. *A function $c_k(\theta)$ intersects a function $l_m(\theta)$ at most twice. Also, functions $c_k(\theta)$ and $u_m(\theta)$ intersect at most twice.*

Proof. By symmetry we can show only that functions $c_k(\theta)$ and $u_m(\theta)$ (or equivalently $2u_m(\theta)$) intersect at most twice. Let (x, y) be the center of the circle $C_k(\theta)$. Then

$$c_k^2(\theta) = (x - x_k)^2 + (y - y_k)^2, \tag{1}$$

$$2u_m(\theta) = (x_{p_m} - x_{p_j})\cos\theta + (y_{p_m} - y_{p_j})\sin\theta. \tag{2}$$

The distance from (x, y) to the lines $r_i(\theta)$ and $s_j(\theta)$ is equal to $c_k(\theta)$

$$c_k(\theta) = (x_{p_i} - x)\sin\theta + (y - y_{p_i})\cos\theta, \tag{3}$$

$$c_k(\theta) = (x - x_{p_j})\cos\theta + (y - y_{p_j})\sin\theta. \tag{4}$$

Using Eq. (2) and $c_k(\theta) = 2u_m(\theta)$, Eqs. (3) and (4) can be written as

$$(x_{p_i} - x - y_{p_m} + y_{p_j})\sin\theta + (y - y_{p_i} - x_{p_m} + x_{p_j})\cos\theta = 0, \tag{5}$$

$$(x - x_{p_j} - x_{p_m} + x_{p_j})\cos\theta + (y - y_{p_j} - y_{p_m} + y_{p_j})\sin\theta = 0. \tag{6}$$

This implies

$$(x_{p_i} - x - y_{p_m} + y_{p_j})(x - x_{p_j} - x_{p_m} + x_{p_j}) = (y - y_{p_i} - x_{p_m} + x_{p_j})(y - y_{p_j} - y_{p_m} + y_{p_j}). \tag{7}$$

Combining Eqs. (1) and (7), we obtain

$$a_1x + a_2y + a_3 = c_k^2(\theta), \tag{8}$$

where a_1, a_2, and a_3 are constants. We assume that in the coordinate system chosen the line $a_1x + a_2y + a_3 = 0$ is horizontal. Thus $a_1 = 0$ and $a_2 \neq 0$. We write

$$y = b_1c_k^2(\theta) + b_2, \tag{9}$$

where $b_1 = 1/a_2$ and $b_2 = -a_3/a_2$. Then Eqs. (5) and (6) can be written as

$$x\sin\theta = A_1\sin\theta + A_2\cos\theta + b_1c_k^2(\theta)\cos\theta, \tag{10}$$

$$x\cos\theta = A_3\sin\theta + A_4\cos\theta - b_1c_k^2(\theta)\sin\theta. \tag{11}$$

By finding $x\sin\theta\cos\theta$ from Eqs. (10) and (11), we obtain

$$A_2\cos^2\theta - A_3\sin^2\theta + (A_1 - A_4)\sin\theta\cos\theta + b_1c_k^2(\theta) = 0. \tag{12}$$

By Eq. (2) and $c_k(\theta) = 2u_m(\theta)$, we have

$$B_1\cos^2\theta + B_2\sin\theta\cos\theta + B_3\sin^2\theta = 0, \tag{13}$$

where B_1, B_2, and B_3 are constants. Eq. (13) has at most two solutions since the domain is $(\theta_0, \theta_0 + \pi/2)$. $\square$

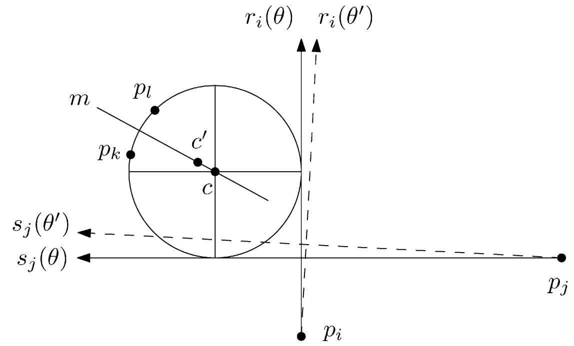

Fig. 11. The points p_k and p_l lie on the second quadrant of the circle.

Theorem 7. *Any two functions in $\mathcal{A}$ intersect at most twice.*

Proof. By Lemma 4 a function $c_k(\theta)$ intersects a function $l_m(\theta)$ (or $u_m(\theta)$) at most twice. It remains to consider two cases.

First, we consider two functions $c_k(\theta)$ and $c_l(\theta)$. We suppose $c_k(\theta) = c_l(\theta)$ and prove that there is not another intersection angle. The siphon types in Fig. 9 imply that the points p_k and p_l can be only in one (the second) quadrant of the circle with center c (Fig. 11). The center c lies on the bisector m of the segment $\overline{p_k p_l}$. Suppose that the functions c_k and c_l intersect for $\theta' \neq \theta$. Without loss of generality $\theta' < \theta$. We decrease θ (rotate both rays clockwise) to θ' and find a point c' on m at the same distance from p_k and $s_j(\theta')$. The point c' is to the left of c on m (Fig. 11). Then the distance from c' to p_k is smaller than $c_k(\theta)$ but the distance from c' to $r_i(\theta')$ is greater than $c_k(\theta)$. Contradiction.

Second, we show that two functions $l_m(\theta)$ and $u_k(\theta)$ intersect at most once. Without loss of generality p_i is at the origin and p_j is on the x-axis (Fig. 10). Then $2l_m(\theta)$ and $2u_k(\theta)$ can be expressed as

$$2l_m(\theta) = x_{p_m} \cos\left(\theta + \frac{\pi}{2}\right) + y_{p_m} \sin\left(\theta + \frac{\pi}{2}\right),$$
$$2u_k(\theta) = (x_{p_k} - x_{p_j}) \cos\theta + y_{p_k} \sin\theta.$$

Then the equation $l_m(\theta) = u_k(\theta)$ can be written as: $a\cos\theta + b\sin\theta = 0$, where a and b are constants. This equation has at most one solution in $(0, \frac{\pi}{2})$. Similar argument can be applied for two functions $l_m(\theta)$ and $l_k(\theta)$. Thus, the theorem follows. □

Let $\mathcal{L}$ be the lower envelope of the arrangement $\mathcal{A}$. By Theorem 7 we can compute $\mathcal{L}$ in $O(n \log^2 n)$ time and $O(n \log n)$ space (Theorem 6.5 in [20]). Thus, by making a line sweep of $\mathcal{L}$ from left to right, we can identify the vertex in $\mathcal{L}$ with the biggest y-coordinate which θ-coordinate gives us the orientation θ of the widest $\frac{\pi}{2}$-siphon supported by p_i and p_j. We store the following information: width, orientation, and critical points defining the $\frac{\pi}{2}$-siphon. Doing this computation for each pair of points in P and updating the widest $\frac{\pi}{2}$-siphon, we can compute the widest arbitrarily-oriented $\frac{\pi}{2}$-siphon in $O(n^3 \log^2 n)$ time and $O(n \log n)$ space.

Remarks.

1. An easy adaptation of the approach above let us to solve the problem for a fixed angle α, $0 < \alpha < \pi$, within the same running time.
2. For $\alpha = \pi$, it corresponds to the widest (arbitrarily-oriented) corridor problem, which has been solved by Houle and Maciel with $O(n^2)$ running time [12].
3. For $\alpha = 0$, the 0-siphon has to produce a non-trivial bipartition of P. Thus, we can push the optimal 0-siphon until the endpoint of its ray contains a point p_i of P. In this case, the ray can contain an extra point p_j of P. The widest arbitrarily-oriented 0-siphon can be computed in $O(n^2 \log n)$ as follows:
   (a) If there is only a point p_i on the ray (on its endpoint), we use the $O(n \log n)$ time algorithm of Follert et al. [9] for computing the widest silo for $P \setminus \{p_i\}$ anchored at p_i, for all $p_i \in P$, and update the widest one.

(b) If the ray contains an extra point p_j of P, the approach used by Follert et al. [9] can be easily modified to solve this case. Let r_{ij} be the ray anchored at p_i and passing through p_j and let $\{\dots, p_{j-1}, p_j, p_{j+1}, \dots\}$ be the ordered sequence of points visited by sweeping the lower envelope of the arrangement of curves defined by the point-ray distance functions. We have that the minimum distance from $P \setminus \{p_i, p_j\}$ to r_{ij} is given by $\min\{d(p_{j-1}, r_{ij}), d(p_{j+1}, r_{ij})\}$. Thus, we compute in $O(n \log n)$ time the widest silo anchored at p_i and passing through every point $p_j \neq p_i$.

By the discussion above we have the following theorem.

Theorem 8. *Given a value α, $0 \leqslant \alpha \leqslant \pi$, the widest arbitrarily-oriented α-siphon for P can be computed in* $O(n^3 \log^2 n)$ *time and* $O(n \log n)$ *space.*

Cheng [2] computes the widest L-shaped corridor in $O(n^3)$ time and $O(n^3)$ space. It will be interesting to know whether the widest arbitrarily-oriented α-siphon with fixed α can be computed in $O(n^3)$ time. In such case, the time complexities of both problems will match.

3.1. Lower bound

We can modify the construction in Theorem 6 as follows: construct a square of side $x_n - x_1$ and make a copy on the instance point set $\{x_1, \dots, x_n\}$ on each of the four sides of the square. Let P be the set of these $4n - 4$ points. The widest arbitrarily-oriented α-siphon through P will be an orthogonal $\frac{\pi}{2}$-siphon passing through the maximum gap of the points in two adjacent sides. Therefore, the $\Omega(n \log n)$ time lower bound in the algebraic decision tree model follows.

4. Anchored siphons

Now we consider a constrained version of the arbitrarily-oriented α-siphon problem. More precisely, the constraint consists on anchoring the corner of the 1-corner polygonal chain at a given point. In the facility location area, this problem asks for the computation of an obnoxious double-ray anchored at a given point.

Without loss of generality, we assume that the anchored point is at the origin O of the coordinate system. By siphon(P, r_1, r_2, w) we denote an α-siphon for P defined by two rays r_1 and r_2 from O, where α is the smallest angle formed by r_1 and r_2, and w is the siphon width. By silo(P, r, w) we denote a silo for P defined by a ray r from O with width w [9]. Obviously, siphon$(P, r_1, r_2, w) = $ silo$(P, r_1, w) \cup$ silo(P, r_2, w) (Fig. 12).

We consider the problems of computing the *widest anchored siphon*, where α is not fixed, and the *widest anchored α-siphon*, where $\alpha \in (0, \pi)$ is fixed.

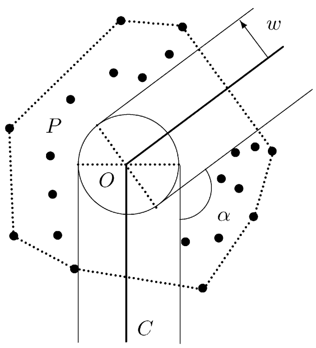

Fig. 12. Anchored and arbitrarily-oriented α-siphon.

4.1. The widest anchored siphon

Let $s_1 = \text{silo}(P, r_1, w_1)$ be the widest anchored silo for P. We define the second widest anchored silo as follows. Let $\gamma_1, \gamma_2, \ldots, \gamma_n$ be the sorted sequence of slopes of rays from O through the points of P. There are n slope intervals (γ_i, γ_{i+1}) (we assume that $\gamma_{n+1} = s_1$). Let w_i^*, $i = 1, \ldots, n$, be the widest anchored silo among all silos with slope in the ith slope interval. Clearly, w_1 is the largest width w_i^*, i.e., $w_1 = \max\{w_i^*\}$. Let w_2 be the second largest width w_i^* and let $s_2 = \text{silo}(P, r_2, w_2)$ be the corresponding silo called the *second widest anchored silo for P*. Let $s = \text{siphon}(P, r_1, r_2, \alpha, w)$ be the anchored siphon for P defined by the rays r_1, r_2 which form an angle α, with $w = \min\{w_1, w_2\}$.

Lemma 5. *If s_1 and s_2 are the widest and the second widest anchored silo for P, respectively, then $s = s_1 \cup s_2$ is the widest anchored siphon for P.*

Proof. The width of s is w. It suffices to show that the width of any anchored siphon is at most w. Let $s' = \text{siphon}(P, r_1', r_2', \beta, w')$ be an anchored siphon. The slopes of r_1' and r_2' are in different slope intervals and one of them, say r_2', is in the interval different from the interval of r_1. Then the width of s' is at most the width of the silo with r_2' which is at most $w_2 = w$. The lemma follows. □

We use Lemma 5 for computing the widest anchored siphon for P.

1. Compute the slopes $\gamma_1, \gamma_2, \ldots, \gamma_n$.
2. Use the algorithm in [9] to compute the widest anchored silo s_1 for P. Let r_1 and w_1 be its ray and width, respectively. The algorithm consists of making a walk along a lower envelope of a set of algebraic functions and finding the highest vertex on it (see [9] for more details). Find i such that the slope of r_1 is in (γ_i, γ_{i+1}).
3. Use again the algorithm in [9] to compute a second widest anchored silo s_2 defined by a ray r_2 with slope in $(\gamma_{i+1}, \gamma_i + 2\pi)$. Let w_2 be its width. The algorithm does a similar walk as in the previous step. The only difference is that the interval (γ_i, γ_{i+1}) is omitted.
4. Construct $\text{siphon}(P, r_1, r_2, w)$ where $w = w_2$.

By Lemma 5, the algorithm computes the widest siphon anchored at O. The steps 1, 2 and 3 can be performed in $O(n \log n)$ time [9] and the step 4 can be done in $O(1)$ time. So, the total running time of the algorithm is $O(n \log n)$.

4.2. The widest anchored α-siphon

For a fixed α, $0 < \alpha < \pi$, there always exists a solution, but the algorithm for widest anchored siphon can not be used since the angle between the widest and the second widest anchored silos may be different from the given α. We show that the problem can be reduced to finding the widest anchored silo. Let P_α ($P_{-\alpha}$) be the set of points obtained by rotating clockwise (counterclockwise) the points of P by the fixed angle α.

Lemma 6. *There exists an anchored α-siphon* $\text{siphon}(P, r_1, r_2, w)$ *for P, where r_2 is the ray r_1 rotated clockwise by α, if and only if there exists an anchored silo $s_2 =$* $\text{silo}(P \cup P_\alpha, r_2, w)$. *Furthermore, the widest anchored silo for $P \cup P_\alpha$ causes the widest anchored α-siphon for P and vice versa.*

Proof. Consider the anchored α-siphon $\text{siphon}(P, r_1, r_2, w)$ for P. The silo s_2 contains no points of P_α since the silo $s_1 = \text{silo}(P, r_1, w)$ contains no points of P. Therefore s_2 contains no points of $P \cup P_\alpha$, see Fig. 13. The converse claim can be shown similarly. □

We use Lemma 6 to find the widest anchored α-siphon for P. Note that the set $P \cup P_{-\alpha}$ can be used instead of $P \cup P_\alpha$ (then the corresponding siphon can be found by rotation the silo clockwise).

1. In $O(n)$ time, compute the $2n$ point set $P \cup P_\alpha$ for the given α.

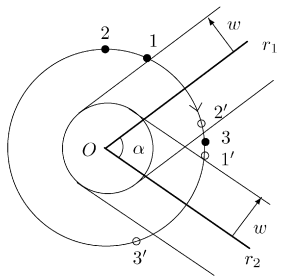

Fig. 13. Illustration of Lemma 6.

2. Use the algorithm in [9] to compute the widest anchored silo s for $P \cup P_\alpha$. Let r and w be its ray and width, respectively. The algorithm makes a walk along the lower envelope, $L_{\mathcal{P}\cup\mathcal{P}_\alpha}$, of a set of algebraic functions l_p, $p \in P \cup P_\alpha$ and find the highest vertex on it. This lower envelope can be computed in $O(n \log n)$ time.
3. Construct siphon(P, r, r', w), where r' is the ray r rotated counterclockwise by the angle α.

Remarks.

1. For $\alpha = \pi$, it corresponds to the widest anchored corridor (cylinder) problem which has been solved by Follert et al. [9] in $\Theta(n \log n)$ time.
2. For $\alpha = 0$, the widest anchored arbitrarily-oriented 0-siphon can be computed in $O(n \log n)$ time as follows: Assume that the anchoring point is p.
   (a) If $p \notin P$, the ray from p has to contain a point $p_i \in P$ according to the non-trivial bipartition condition. Let ℓ be the line passing through p and p_i. Using the Voronoi diagram, in $O(\log n)$ time compute the nearest neighbor of p, say p_1, and $d_1 = d(p, p_1)$; compute the nearest neighbor of p_i, say p_2, $d_2 = d(\ell, p_2)$. The width is $\min\{d_1, d_2\}$.
   (b) If $p \in P$ and the ray from p only contains point p, we solve the problem in $O(n \log n)$ time applying the Follert et al. [9] algorithm of the widest silo for $P \setminus \{p\}$ anchored at p. If the ray from p contains a point p_j, we proceed as above.

By the discussion above we conclude the following theorem.

Theorem 9. *The widest anchored α-siphon problem for P with fixed or unfixed α, $0 \leqslant \alpha \leqslant \pi$, can be solved in $O(n \log n)$ time.*

4.3. Lower bound

Next we show an $\Omega(n \log n)$ time lower bound for the widest anchored and arbitrarily-oriented α-siphon problem. We distinguish two different cases.

Unfixed α. Let $P = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be an instance of the MAX-GAP problem for points on the first quadrant of the unit circle centered at the origin O of the coordinates system [14,19]. Compute the widest anchored siphon $s = \text{siphon}(P, r_1, r_2, w)$ for P anchored at O. Let p (q) be the intersection point of the ray r_1 (r_2) with the unit circle. Let a and b (c and d) be the first point in P after p (q) in the counterclockwise and clockwise order, respectively (Fig. 14). These points can be computed in $O(n)$ time. Compute the distances $d(a, b)$ and $d(c, d)$.

Because we find a non trivial partition, at most one of the rays pass through P. Notice that both r_1 and r_2 of an optimal siphon pass through P if and only if $d(a, b) = d(c, d)$. Otherwise, the siphon is not optimal since another improved siphon can be found by considering a ray (the corresponding to $\min\{d(a, b), d(c, d)\}$) outside the quadrant. As a consequence, the distance $\min\{d(a, b), d(c, d)\}$ defines the maximum gap for P. As the maximum gap for points on the first quadrant of the unit circle has an $\Omega(n \log n)$ lower bound in the algebraic decision tree model [14,19], the result follows.

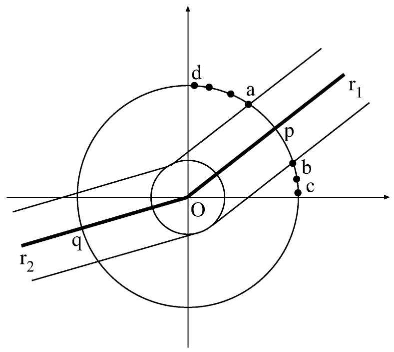

Fig. 14. Lower bound for the widest anchored siphon.

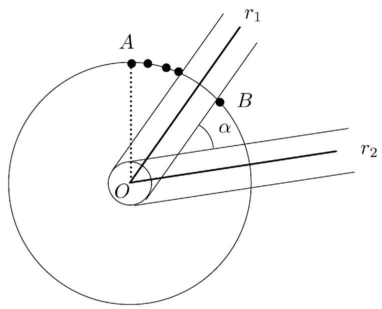

Fig. 15. Lower bound for the widest anchored α-siphon.

Fixed α. The MAX-GAP problem for points in a quadrant on the unit circle has an $\Omega(n \log n)$ time lower bound in the algebraic decision tree model [14,19]. For points in a given arc AB on the upper-half of the unit circle the same proof can be adapted straightforward.

Let AB be an arc on the upper half of the unit circle corresponding to the given angle α, and let $P = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be an instance of the MAX-GAP problem for points in the arc AB. In $\mathrm{O}(n)$ time compute the minimum and the maximum of P in their x-coordinate order. Without loss of generality, we can suppose that $A = \min(P)$ and $B = \max(P)$. Compute the widest anchored α-siphon for P anchored at O which produces a non-trivial bipartition of P (this is guaranteed by the algorithm above) which implies that one of the silos of the α-siphon pass through P (Fig. 15). Let r_1 and r_2 be the two rays of the computed α-siphon. By construction the silo anchored in O which pass through P defines the maximum gap between consecutive points of P, because otherwise the computed α-siphon is not the widest one. Let p_1 and p_2 be the intersection points of r_1 and r_2 with the unit circle, respectively. In $\mathrm{O}(n)$ time compute the two closer points a_1 and b_1 (a_2 and b_2) of P to p_1 (p_2). The distance $\min\{d(a_1, b_1), d(a_2, b_2)\}$ defines the maximum gap for P.

By the discussion above we can conclude the following theorem.

Theorem 10. *The widest anchored siphon and the widest anchored α-siphon for P can be computed in optimal $\Theta(n \log n)$ time.*

5. Conclusion

In this paper, we introduce a new kind of empty corridor by considering a curved boundary in the shape of the corridor. Such corridor, the α-siphon, corresponds to the locus of points at distance w from a 1-corner polygonal chain with interior angle α. We present algorithms for finding a widest α-siphon for both the oriented and the arbitrarily-oriented case. In the latter case, a constrained version is also considered that arises in the computation of an obnoxious anchored 1-corner path in facility location. For this case, we present an optimal algorithm even when the interior angle α is considered as a new variable in the problem.

References

[1] S.-W. Chen, Widest empty corridor with multiple links and right-angle turns, in: Proc. 6th Canadian Conference on Computational Geometry, 1994, pp. 57–62.
[2] S.-W. Chen, Widest empty L-shaped corridor, Information Processing Letters 58 (1996) 277–283.
[3] K. Daniels, V. Milenkovic, D. Roth, Finding the largest area axis-parallel rectangle in a polygon, Computational Geometry: Theory and Applications 7 (1997) 125–148.
[4] J.M. Díaz-Báñez, F. Hurtado, Computing obnoxious 1-corner polygonal chains, Computers and Operations Research 33 (2006) 1117–1128.
[5] J.M. Díaz-Báñez, F. Hurtado, H. Meijer, D. Rappaport, J.A. Sellarès, The largest empty annulus problem, International Journal of Computational Geometry and Applications 13 (2003) 317–325.
[6] J.M. Díaz-Bañez, C. Seara, I. Ventura, The siphon problem, in: 20th European Workshop on Computational Geometry, 2004, pp. 33–36.
[7] J.M. Díaz-Bañez, M.A. López, J.A. Sellarès, Finding a widest empty 1-corner corridor, Information Processing Letters 98 (2006) 199–205.
[8] J.M. Díaz-Báñez, J.A. Mesa, A. Shöbel, Continuous location of dimensional structures, European Journal of Operational Research 152 (2004) 22–44.
[9] F. Follert, E. Schömer, J. Sellen, M. Smid, C. Thiel, Computing the largest empty anchored cylinder, and related problems, International Journal of Computational Geometry and Applications 7 (1997) 563–580.
[10] A. Glozman, K. Kedem, G. Shpitalnik, Computing a double-ray center for a planar point set, International Journal of Computational Geometry and Applications 9 (1999) 109–124.
[11] T.L. Heath, Apollonius of Perga: Treatise on Conic Sections, W. Heffer and Sons Ltd., Cambridge, UK, 1896.
[12] M.E. Houle, A. Maciel, Finding the widest empty corridor through a set of points, in: G. Toussaint (Ed.), Snapshots of Computational and Discrete Geometry, Technical Report SOCS-88.11, School of Computer Science, McGill University, 1988.
[13] H.T. Kung, F. Luccio, F.P. Preparata, On finding the maxima of a set of vectors, Journal of ACM 22 (1975) 469–476.
[14] D.T. Lee, Y.F. Wu, Geometric complexity of some location problems, Algorithmica 1 (1986) 193–211.
[15] N. Megiddo, Applying parallel computation algorithms in the design of serial algorithms, Journal of ACM 30 (4) (1983) 852–865.
[16] A. Mukhopadhyay, S.V. Rao, On computing a largest empty arbitrarily oriented rectangle, International Journal of Computational Geometry and Applications 13 (2003) 257–371.
[17] S.C. Nandy, B.B. Bhattacharya, On finding an empty staircase polygon of largest area (width) in a planar point-set, Computational Geometry: Theory and Applications 26 (2003) 143–171.
[18] F.P. Preparata, M.I. Shamos, Computational Geometry, An Introduction, Springer-Verlag, 1988.
[19] V. Sacristán, Lower bounds for some geometric problems, Technical Report MA-IR-98-0034, Departament de Matemàtica Aplicada II, Universitat Politècnica de Catalunya, 1998.
[20] M. Sharir, P.K. Agarwal, Davenport–Schinzel Sequences and their Geometric Applications, Cambridge University Press, 1995.
[21] G.T. Toussaint, Computing largest empty circles with location constraints, International Journal of Computer and Information Sciences 12 (1983) 347–358.