The siphon problem

J.M. Díaz-Báñez [*,a,1], C. Seara [b,2] and I. Ventura [a,1]

[a] *Universidad de Sevilla, Spain*

[b] *Universitat Politècnica de Catalunya, Spain*

[c] *Universidad de Huelva, Spain*

Abstract

An α-siphon is the locus of points in the plane that are at the same distance ϵ from a polygonal chain consisting of two half-lines emanating from a common point such that α is the interior angle of the half-lines. Given a set S of n points in the plane and a fixed angle α, we want to compute an α-siphon of largest width ϵ such that no points of S lies in its interior. We present an efficient $O(n^2)$-time algorithm for computing an orthogonal siphon. The approach can be handled to solve the problem of the oriented α-siphon for which the orientation of a half-line is known. We also propose an $O(n^3 \log n)$-time algorithm for the arbitrarily oriented version.

1. Introduction

A corridor through a planar point set S is the open region of the plane that is bounded by two parallel lines intersecting the convex hull of S, $CH(S)$. A corridor is empty if it does not contain any point of S. The problem of computing the widest empty corridor through a set S of n points in the plane has been solved by Houle and Maciel [3] in $O(n^2)$ time (Figure 1a).

One of the possible motivations of the widest empty corridor problem is to find a collision-free route to transport objects through a set of point obstacles. However, even the widest empty corridor may not be wide enough sometimes. This motivates to consider allowing right-angle turns. Chen in [1] 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).

* Corresponding author

Email addresses: dbanez@us.es (J.M. Díaz-Báñez), carlos.seara@upc.es (C. Seara), inmaculada.ventura@dmat.uhu.es (I. Ventura).

[1] The first author is partially supported by Project MCYT BFM2000-1052-C02-01.

[2] The second author is partially supported by projects MCYT-FEDER TIC-2001-2171, MCYT-FEDER BFM 2002-0557, Gen Cat 2001SGR00224.

In this paper we consider a kind of corridor problem which we call the *siphon problem.* More precisely, we define a *siphon* as the locus of points in the plane that are at the same distance ϵ from a polygonal chain P consisting of two half-lines emanating from a common point (a 1-corner polygonal chain). Let α be the interior angle of the half-lines. An α-siphon is defined similarly but with the constraint that the interior angle of the two half-lines emanating from a common point is α.

An α-siphon is determined by P and ϵ, where ϵ is called the *siphon width.* Possible values for the *siphon angle* α are $0^\circ \leq \alpha \leq 180^\circ$. The α-siphon problem can be stated as follows.

α-Siphon problem. *Given a set S of n points in the plane and a fixed angle α, compute the α-siphon of largest width such that no points $p \in S$ lies in its interior (Figure 1b).*

The α-siphon has to intersect the convex hull of S producing a non-trivial partition S_1, S_2 of S; otherwise we will allow the α-siphon "to scratch the exterior" of S without actually passing through S and, therefore, the α-siphon can be arbitrarily wide (Figure 1c).

Notice that a 180°-siphon is just a corridor [3]. A 0°-siphon is a *silo* emanating from a point (the endpoint of a half-line) (Figure 1d) and it has been partially studied in [2] by giving an optimal $\Theta(n \log n)$-time algorithm in the case that the endpoint of the half-line is anchored on a given point.

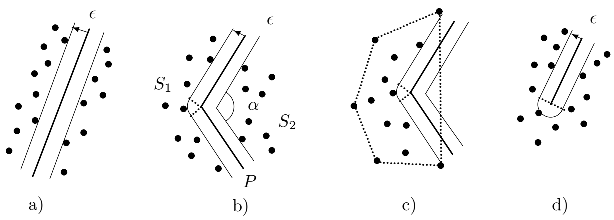

Fig. 1. a) Widest corridor, b) α-siphon c) unbounded width siphon, c) silo

Notice also that our α-siphon is a kind of corridor which is a "better" solution than Cheng's corridor in the following sense: suppose that we are interesting in transporting a circular object in between a set of points, Cheng's algorithm can gives a negative answer while our algorithm produces 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 a siphon is the area swept by a disk whose center describes the route.

In this paper two variants for the α-siphon problem are consider: i) the *oriented α-siphon problem*, where we know the angle α and the direction of one of the half-lines of P; ii) the *arbitrarily oriented α-siphon problem*, where only the angle α is known. Most proofs are omitted in this extended abstract.

Let $S = \{p_1, p_2, \ldots, p_n\}$ be a planar point set. The points are in general position, this assumption is not essential for our algorithms to work, but handling degeneracies would require the description of many details and would hide the crucial ideas. We denote the Euclidean distance between two points p and q by $d(p, q)$. If p is a point and P is a closed subset in the plane, the distance between p and P is defined as $d(p, P) = \min\{d(p, q) : q \in P\}$.

An α-siphon is bounded by an outer boundary and an inner boundary; the outer boundary is formed by a circular arc joined to two half-lines, the *exterior boundary legs*; and the inner boundary is formed by two half-lines, the *interior boundary legs*. By *orthogonal siphon* we denote an oriented 90°-siphon such that its boundary legs are vertical and horizontal.

2. Oriented α-siphon

In this section we study the problem of computing an oriented α-siphon. First we consider the orthogonal siphon and next we will consider the general case. There are four possibilities for P in an orthogonal siphon according to the north, south, east and west directions. We only consider the case S-E, other cases can be handle analogously.

First notice that for any 1-corner polygonal chain P which does not contain points from S and produces a non-trivial partition of S, always exists a siphon defined by P. We call a siphon *non-expansive* if its interior boundary contains two points of S (one per each leg or only one point if this point is on the vertex of that boundary) and its exterior boundary contains one point of S.

Lemma 1 *For any fixed vertical and horizontal lines crossing the convex hull of S producing a non trivial partition of S, there exists a non-expansive siphon.*

Next we describe the algorithm which solves the S-E orthogonal siphon problem in $O(n^2)$ time. By Lemma 1 we know that an orthogonal siphon is defined by at most three points. First, the algorithm sorts the points in S by decreasing y-coordinate and by increasing x-coordinate in $O(n \log n)$ time. Let $O = \{p_1, \ldots, p_n\}$ and $A = \{q_1, \ldots, q_n\}$ be the respective lists of the sorted points. From a vertical-horizontal grid we construct a S-E staircase E which is updated each time we insert a new point, and in this case either the number of the steps of E increase by one or it decrease because the new point *dominates* some points of the current E.

Let p_1 and q_1 be the points of S with maximum y-coordinate and minimum x-coordinate, respectively. These two points determine the starting point of the algorithm. The staircase E in the initial stage is formed by the horizontal and vertical half-lines of the grid passing through p_1 and q_1. The algorithm compute all the possible orthogonal siphons which horizontal boundary leg is supported by $p_i \in O$, for $i = 2, \ldots, n$. If a point $p_i = (x_{pi}, y_{pi})$ is on the horizontal-interior boundary leg of a siphon then, there only exist siphons such that its "entries" are in-between points having x-coordinates and y-coordinates smaller than or equal to x_{pi} and y_{pi}, respectively; because the rest of points either are *dominated* by either the current staircase or the point p_i. The dominance relation between points of S is establish as in [4,5]. In [5] the maxima problem for a set of points is considered. The maxima problem consists of finding all the maxima of S under dominance and they can be computed in $\theta(n \log n)$ time. We are interested in the maxima problem with the following dominance relation: $p_j \prec p_i \iff x_{pj} \leq x_{pi}$ *and* $y_{pj} \geq y_{pi}$. The corresponding set of maxima forms a *four quadrant staircase.*

We construct the staircase E in an incremental way with a cost of $O(\log n)$ time per point insertion. In the worst case the number of different siphons to be checked can be $O(n^2)$. Assume that O and A have been computed and also for each point p_i we have a pointer to q_l such that $p_i = q_l$.

ORTOGONAL-SIPHON-ALGORITHM
Input: S, O, A,
Output: Widest orthogonal siphon,

(i) **Initial stage:** O, A, $E :=$ staircase formed by the horizontal and vertical half-lines defined by q_1 and p_1. Compute the Voronoi diagram for S, $VD(S)$, and store it in a data structure such that a query point can be answered in $O(\log n)$ time.

(ii) **For** $i = 2$ to n, **do** "Compute the widest orthogonal siphon supported by p_i and q_j, such that $x_{q_j} < x_{p_i}$, $y_{q_j} < y_{p_i}$",

Let ϵ_1 be the distance between $y = y_{pi}$ and the horizontal line containing the segment of the current E which is intersected by $x = x_{qj}$. Compute $\epsilon_2 = x_{qj} - x_{q(j-1)}$ and $\epsilon_0 = \min\{\epsilon_1, \epsilon_2\}$. The orthogonal siphon supported by p_i and q_j has width smaller than or equal to $\epsilon_0/2$. Compute the part E_{ij} of E in-between the x-coordinates $x_{qj} - \epsilon_0$ and x_{qj} and the y-coordinates y_{pi} and $y_{pi} + \epsilon_0$. Check that E_{ij} is empty and compute the orthogonal siphon of width $\epsilon_0/2$ supported by p_i and q_j. Otherwise, we analyze each point of E_{ij} determining (if it exists) the corresponding orthogonal siphon as follows:

The vertex of the 1-corner polygonal line of the siphon will be located in the bisector of the second quadrant passing through (x_{qj}, y_{pi}). This vertex is the center of the circle which contains the arc of the siphon. By using E and $VD(S)$, we consider each of the three possible locations for the point belonging to the exterior boundary.

(iii) **Updating stage:** Insert p_i in E and update E in time $O(\log n)$ per insertion and in total time $O(n \log n)$ for the deletion of points from E (a point is deleted only once). Delete p_i from O, delete $q_l = p_i$ from A, update the widest width and the current siphon.

Lemma 2 *Each vertex c_i of the staircase E is analyzed at most once.*

Analysis of the algorithm: The number of stages is $O(n^2)$. At the stage (i, j) we check all the vertices in E_{ij} in $O(\log n)$ time. By Lemma 2 a vertex in E is analyzed only once, then the total time cost in the analysis of points in E is $O(n \log n)$. Therefore the running time of the algorithm is $O(n \log n) + O(n^2) = O(n^2)$.

Theorem 3 *The widest orthogonal siphon can be computed in $O(n^2)$ time.*

The techniques above can be adapted for computing the widest oriented α-siphon, i.e., a α-siphon with a given angle α, $0 < \alpha < 180°$ and a fixed direction of one of its half-lines.

Corollary 4 *The widest oriented α-siphon can be computed in $O(n^2)$ time.*

Theorem 5 *The problem of computing the widest oriented α-siphon has an $\Omega(n \log n)$ time lower bound in the algebraic decision tree model.*

Notice that the complexity of the algorithm above match the complexity of the algorithm for computing the widest corridor of a set of points. A small variation of the algorithm above can be used to compute the widest L-shaped orthogonal corridor (as defined by Chen [1]) with the same $O(n^2)$ running time.

Corollary 6 *The widest L-shaped orthogonal corridor can be computed in $O(n^2)$ time.*

A similar algorithm can be used if we want to compute a corridor of the same kind but with an angle different from 90° and knowing the direction of one of the links.

3. The arbitrarily-oriented α-siphon

In this section we deal with the computation of a widest-empty arbitrarily-oriented α-siphon, i.e., we only fix the *siphon angle* α. Assume that α is $\frac{\pi}{2}$.

Lemma 7 *There always exists an optimal $\frac{\pi}{2}$-siphon such that the interior boundary contains two points of S (one per each leg) or only one point if this point is on the corner of that boundary.*

The points in S that determine a tentative placement of an optimal α-siphon are called the *critical points*. Therefore we can classify the cases for critical points according to their location on the parts of the siphon, as it is shown in Figure 2.

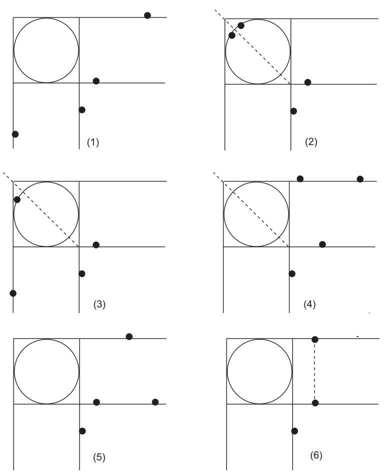

Fig. 2. Types of candidate siphons.

We sketch the idea of our approach without details. We only consider siphons that are bounded by a point of each its interior half-lines (according to Lemma 7). The orientation θ of our siphon is the smaller of the two angles given by the ortogonal lines supporting the interior boundary legs. Given two points p_i, p_j ($x_{p_i} > x_{p_j}$), we consider the rotation of the two perpendicular lines $r_i(\theta)$ (around p_i) and $s_j(\theta)$ (around p_j), say counterclockwise, to obtain all possible empty siphon supported at p_i and p_j. The essence of our algorithm is to generate a discrete set of subintervals in such rotation and compute a siphon of maximum width for each.

We begin the rotation in $\theta = 0$. A pair of orthogonal lines through p_i and p_j partitions the point set into four disjoint subsets which we label I, II, III and IV (corresponding to the four quadrants). As we change the orientation continuously, the partition survive till some two points become collinear. In fact, by insertion and deletion of the points, we can maintain dynamically each one of the corresponding partition. This spend $O(n^2)$ time and space. This produces a partition of the rotation interval. Taking into account the intervals of such partitions for each point $p_k \in S$, we define the following functions:

- $u_k(\theta) = d(p_k, r_i(\theta))$, for $p_k \in I$,
- $u_k(\theta) = d(p_k, r_i(\theta))$, $l_k(\theta) = d(p_k, s_j(\theta))$ and $c_k(\theta) = d(p_k, c_{ijk}(\theta))$, for $p_k \in II$,
- $l_k(\theta) = d(p_k, s_j(\theta))$, for $p_k \in III$,

where $c_{ijk}(\theta)$ is the center of the circle passing through p_k tangent to the lines $r_i(\theta)$ and $s_j(\theta)$.

Lemma 8 *Let p_k and p_l be two distinct points of S. Then, the graphs of two functions corresponding to p_k and p_l intersect at most twice.*

Let $\mathcal{L}$ be the lower envelope of the graphs of the functions u_k, l_k and c_k. Lemma 8 implies that the labels of the points corresponding to the edges of $\mathcal{L}$, when we traverse $\mathcal{L}$ from left to right, form a Davenport-Schinzel sequence of order two [6]. Therefore, the number of intervals in the partition is $O(n)$ [6], and by using a standard divide-and-conquer approach we can compute $\mathcal{L}$ in $O(n \log n)$ time. Thus, by traversing $\mathcal{L}$, from left to right, we can identify the highest vertex, which corresponds to the optimal direction for the $\frac{\pi}{2}$-siphon. In summary, working over all pair of points in S, we have proven the following result.

Theorem 9 *Given a set S of n points in the plane, the widest empty arbitrarily-oriented $\frac{\pi}{2}$-siphon can be computed in $O(n^3 \log n)$ time.*

An adaptation of above approach permits to solve the problem for a fixed angle α, $0° \leq \alpha \leq 180°$ in the same time bound.

A constrained version of this problem consists into anchoring the vertex of the 1-corner polygonal chain. In this case we obtain the following result.

Theorem 10 *Given a set S of n points in the plane, the widest-empty arbitrarily-oriented anchored α-siphon can be computed in optimal $\Omega(n \log n)$ time.*

References

[1] S-W. Chen, *Widest empty L-shaped corridor*, Information Processing Letters, 58, 1996, pp. 277–283.

[2] F. Follert, E. Schömer, J. Sellen, M. Smid, C. Thiel, *Computing the largest empty anchored cylinder, and related problems*, Inter. Journal of Comput. Geometry and Aplications, Vol. 7 (6), 1997, pp. 563–580.

[3] M. E. Houle, A. Maciel, *Finding the widest empty corridor through a set of points*, in Snapshots of Computational and Discrete Geometry, Godfried Toussaint, ed., Technical Report SOCS-88.11, School of Computer Science, McGill University, 1988.

[4] H. T. Kung, F. Luccio, F. P. Preparata, *On finding the maxima of a set of vectors*, Journal of ACM, 22(4), 1975, pp. 469–476.

[5] F. P. Preparata, M. I. Shamos, *Computacional Geometry, An introduction*, Springer-Verlag, 1988.

[6] M. Sharir, P. K. Agarwal, *Davenport-Schinzel sequences and their geometric applications*, Cambridge University Press, 1995.