Available online at www.sciencedirect.com

SCIENCE DIRECT

Information Processing Letters 98 (2006) 199–205

Information Processing Letters

www.elsevier.com/locate/ipl

On finding a widest empty 1-corner corridor

J.M. Díaz-Báñez [a], M.A. López [b], J.A. Sellarès [c,*]

[a] Universidad de Sevilla, Spain
[b] University of Denver, USA
[c] Universitat de Girona, Spain

Received 18 April 2005; received in revised form 1 December 2005; accepted 12 February 2006

Available online 9 March 2006

Communicated by S.E. Hambrusch

Abstract

A 1-corner corridor through a set S of points is an open subset of $CH(S)$ containing no points from S and bounded by a pair of parallel polygonal lines each of which contains two segments. Given a set of n points in the plane, we consider the problem of computing a widest empty 1-corner corridor. We describe an algorithm that solves the problem in $O(n^4 \log n)$ time and $O(n)$ space. We also present an approximation algorithm that computes in $O(n \log n/\varepsilon^{1/2} + n^2/\varepsilon)$ time a solution with width at least a fraction $(1-\varepsilon)$ of the optimal, for any small enough $\varepsilon > 0$.

Keywords: Computational geometry; Algorithms; Widest empty corridors

1. Introduction

A *corridor* $c = (\ell, \ell')$ is the open region of the plane bounded by two parallel straight lines ℓ and ℓ'. The width of c is the Euclidean distance $d(\ell, \ell')$ between its two parallel bounding lines. A *link* L is an open region defined by two parallel rays $r(L) = p + \vec{v}t$ and $r'(L) = p' + \vec{v}t$, and an open line segment $s(L) = \overline{pp'}$, forming an unbounded trapezoid. We consider the points of $s(L)$, but not those of $r(L)$ and $r'(L)$, as part of the link. The width of a link L, denoted by $\omega(L)$, is the Euclidean distance $d(r(L), r'(L))$ between its bounding parallel rays.

A 1-*corner corridor* $C = (L, L')$ is the union of two links L and L' sharing only the segment $s(L) = s(L')$. Thus, C is an open region bounded by an *outer boundary* that contains a convex corner with respect to the interior of the corridor, and an *inner boundary* that contains a concave corner. Each boundary consists of two rays which we call the *boundary legs*. We adopt the convention of using $r(L)$ and $r(L')$ (resp. $r'(L)$ and $r'(L')$) to denote the legs of the outer (resp. inner) boundary. The *width* of a 1-corner corridor C, denoted by $\omega(C)$, is the smaller of the widths of its two links. The *angle* $\alpha(C)$, $0 < \alpha(C) \leqslant \pi$, of the 1-corner corridor $C = (L, L')$ is the angle determined by the rays $r(L)$ and $r(L')$.

Let S be a set of n points in the Euclidean plane. A corridor c intersecting the convex hull, $CH(S)$, of S is *empty* if it does not contain any points of S. An empty corridor must intersect $CH(S)$, as otherwise the widest empty corridor is not well-defined. A 1-corner corridor C is *empty* if it does not contain any points

* Corresponding author.
E-mail address: sellares@ima.udg.es (J.A. Sellarès).

0020-0190/$ – see front matter
doi:10.1016/j.ipl.2006.02.002

of S and its removal partitions the plane into two unbounded regions, each containing at least one point of S. Note that, as suggested in [2] (for the case $\alpha(C) = \pi/2$), it no longer suffices to require that candidate corridors intersect $CH(S)$, as this would allow such corridors to "scratch the exterior" of S without really "passing through" it.

Widest empty corridor problems belong to geometric optimization, an active area of research in the field of computational geometry. The placement of empty geometric objects (circle, rectangle, unbounded rectangular strip, annulus, etc.) of "maximum measure" among a set of points have been extensively studied [10,8,6,2,4]. The motivation for computing optimal empty figures comes from a variety of practical problems such as robot manipulation [6], computer-aided design [9] and obnoxious facility location [3].

The problem of computing a widest empty corridor can be solved in $O(n^2)$ time and $O(n)$ space [6,7]. Cheng shows how to compute a widest empty 1-corner corridor for the case of fixed $\alpha(C) = \pi/2$, in $O(n^3)$ time and $O(n^3)$ space [2]. In this paper we allow $\alpha(C)$ to assume arbitrary values and describe an algorithm to compute a widest empty 1-corner corridor in $O(n^4 \log n)$ time and $O(n)$ space. As the solution to this problem may be not unique, we settle for computing one optimal corridor. We also present an algorithm that, for any small enough $\varepsilon > 0$, computes in time $O((n \log n)/\varepsilon^{1/2} + n^2/\varepsilon)$ an empty 1-corner of width at least $(1-\varepsilon)w^*$, where w^* is the width of a widest empty 1-corner corridor. For a fixed ε this time complexity is in $O(n^2)$.

2. Locally widest corridors

Since an empty corridor C can always be considered a 1-corner corridor with $\alpha(C) = \pi$, our optimal 1-corner corridor is at least as wide as the widest empty corridor. Thus, we can dismiss this case in $O(n^2)$ time and concentrate our attention on 1-corner corridors with $0 < \alpha(C) < \pi$.

In the sequel, whenever we talk about a corridor we mean a 1-corner corridor and, unless otherwise specified, we assume that this corridor is empty.

We begin with the obvious observation that there exists an optimal solution C that contains at least one point of S in each leg. Otherwise, the width of one or both links of C can be increased by sliding the two legs of a link apart until a point of S is encountered, a process that does not decrease the width of the corridor.

We say that a link of a corridor is *locally widest* if each leg contains at least one point of S and it is not possible to increase its width by performing rotations of the legs around the points from S incident on the leg boundaries. Any such rotation must, of course, guarantee that the two legs of the link share the same slope. For instance, if a link contains exactly one point on each leg, its width can be increased by rotations if the legs are not orthogonal to the segment joining the two points. Similarly, a corridor is locally widest if both of its links are locally widest. We proceed to characterize locally widest links.

Lemma 1. *A link L is locally widest if and only if it satisfies one of the following conditions*:

(21) *There are points p_1 and p_2 of S that lie on the outer leg $r(L)$ and a point q of S that lies on the inner leg $r'(L)$, such that both $\angle qp_1p_2$ and $\angle qp_2p_1$ are acute.*

(12) *There are points p_1 and p_2 of S that lie on the inner leg $r'(L)$ and a point q of S that lies on the outer leg $r(L)$, such that both $\angle qp_1p_2$ and $\angle qp_2p_1$ are acute.*

(11) *There are points p and q of S that lie on the outer and inner legs of L, respectively, such that $\overline{pq}$ is orthogonal to both $r(L)$ and $r'(L)$.*

Proof. Assume L is locally widest. Then, each leg of L contains at least one point. Assume now that each leg contains exactly one point of S (case 11), say p and q, respectively. If $\overline{pq}$ is not orthogonal to the legs, then both legs can be rotated simultaneously by the same amount around p and q, respectively, so as to increase the link width while keeping the link empty. This contradicts the optimality of L. We consider now the remaining two cases (21 and 12). Let $B = S \cap r(L)$, i.e., the points of S that lie on the outer leg. Similarly, let $B' = S \cap r'(L)$. If there is a line h orthogonal to $r(L)$ (hence, also to $r'(L)$) that separates B from B' then a small simultaneous rotation around the points of each leg closest to h increases the width of L, which is not possible because L is locally widest. One can now see that there must be a point $q \in B \cup B'$ such that the line through q orthogonal to $r(L)$ splits the points of S on the leg not containing z into two non-empty subsets. We let p_1 and p_2 be any two points on opposite sides of h, on the leg not containing q. Clearly, both $\angle qp_1p_2$ and $\angle qp_2p_1$ are acute. Notice that, as expected, no rotation of the links can increase the distance between the legs while leaving the link empty. The proof of the converse is straightforward. □

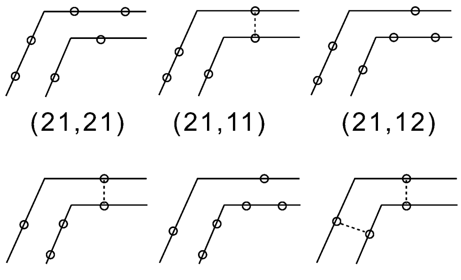

Fig. 1. Locally widest corridors.

From the characterization given in Lemma 1 we obtain only six classes of locally widest corridors, which we label (21, 21), (21, 11), (21, 12), (12, 11), (12, 12), (11, 11), depending on the types of the participating links. The six types of corridors are illustrated in Fig. 1.

Lemma 1 implies that when looking for an optimal solution it suffices to examine locally widest corridors (or a superset). In fact, assume that C is an optimal corridor that is not locally widest, with links of widths $\omega(L)$ and $\omega(L')$. At least one of L and L' must be locally widest as, otherwise, the width of C may be improved, contradicting its optimality. Assume L is locally widest but L' is not. Then, $\omega(L) \leqslant \omega(L')$, and L' can be rotated to increase its width until it becomes locally widest, an operation that does not change the overall width of C. Thus we have the following.

Lemma 2. *There always exists an optimal* 1*-corner corridor that is locally widest.*

Our approach to compute a widest empty corridor consists of systematically generating a superset $\mathcal{C}$ of the locally widest corridors which, by Lemma 2, is guaranteed to contain a solution. Since there are members of $\mathcal{C}$ with six points on the boundary, a brute force algorithm would run in $O(n^7)$ time, as we have $O(n^6)$ candidates and need $O(n)$ time to check each candidate for emptiness. Instead, our algorithm generates one boundary for each link, and computes the remaining boundaries by translating and adjusting a copy of a leg until a point of S is encountered. This allows us to compute an optimal 1-corner corridor in $O(n^4 \log n)$ time.

3. Preliminaries

For any two points p and q we denote by ℓ_{pq} the line through p and q and for any line ℓ and point $t \notin \ell$ we denote by $H_\ell^+(t)$ (resp. $H_\ell^-(t)$) the open halfplane bounded by ℓ that contains (resp. does not contain) t.

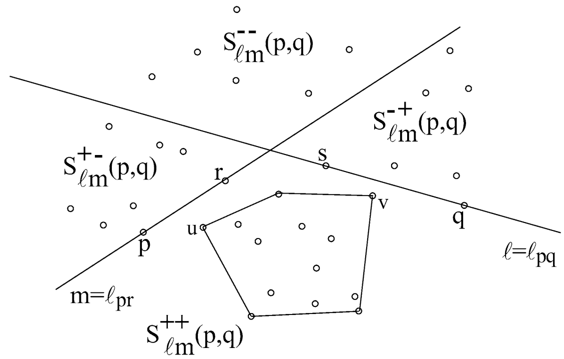

Fig. 2. Partition of the plane by two lines.

When $\ell = \ell_{pq}$, we simply write $H_{pq}^+(t)$ (resp. $H_{pq}^-(t)$). Also for any two non-parallel lines ℓ and m and points p and q, we let $S_{\ell m}^{++}(p,q)$ denote the subset of S in the interior of $H_\ell^+(p) \cap H_m^+(q)$. The other three regions determined by ℓ and m are labeled $S_{\ell m}^{--}(p,q)$, $S_{\ell m}^{-+}(p,q)$, and $S_{\ell m}^{+-}(p,q)$, as illustrated in Fig. 2.

Our algorithms depend on finding efficiently a point of a subset P of S closest to a line ℓ that bounds a halfplane containing P. To this end, we use the following observation.

Observation 1. *Let H be an open halfplane bounded by a line ℓ and let P be a set of m points in H. The point of P closest to ℓ is a vertex of the convex hull $CH(P)$ of P. Once $CH(P)$ has been computed, this point can be found in* $O(\log m)$ *time.*

In the example of Fig. 2, where $P \subset S_{\ell m}^{++}(p,q)$, u and v are the points of P closest to ℓ_{pr} and ℓ_{qs}, respectively. As we shall see, points u, p and r (resp. v, q and s), will be used to define the boundaries of one of the links of a 1-corner corridor.

4. The algorithm

We describe an algorithm to compute a widest 1-corner corridor by processing each of the six cases described in Section 2. To simplify the description we assume that no three points of S are collinear. In practice, collinear degeneracies can be coped by using the simulation of simplicity technique [5].

4.1. *Cases* (21, 21)*,* (21, 11)*,* (21, 12)

All cases in which at least one outer leg contains two points can be handled in a uniform way. Assume that an optimal corridor contains two points p_1, p_2 on one outer leg and one point q_1 on the other. We do not discard the

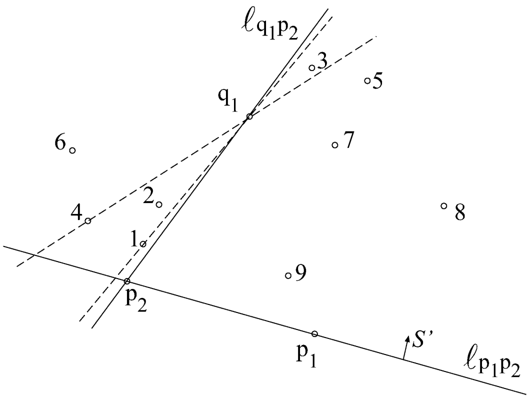

Fig. 3. Computing corridors (21, 21), (21,12), (21,11). Solid line $\ell_{q_1 p_2}$ defines the start of the sweep.

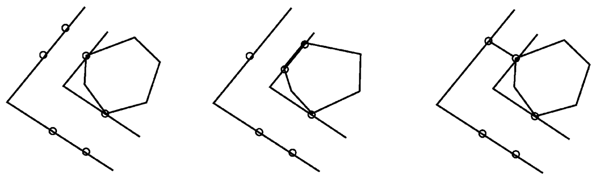

Fig. 4. Events to consider.

possibility that $q_2 = p_2$, which means that the convex corner of the corridor is a point from S that lies on both outer legs. We compute all candidate corridors of types $(21, 21)$, $(21, 11)$ and $(21, 12)$ for each three points q_1, p_1, p_2 of S.

First, for each point $q_1 \in S$, we compute the radial ordering of $S \setminus \{q_1\}$ as a line ℓ through q_1 rotates around q_1. The initial orientation of ℓ is arbitrary and rotation angles in the range $[0, \pi)$ guarantee that each input point is visited exactly once. In practice, depending on the type of corridor sought, only a sublist of the entire sorted list will be needed, but this sublist depends of the choice of p_1 and p_2.

Now, given two points p_1, p_2 from $S \setminus \{q_1\}$, let $m = \ell_{p_1 p_2}$ and $S' = S \cap H_m^+(q_1)$. The idea is to consider all corridors that contain p_1 and p_2 on one outer leg, and q_1 on the other, and keep track of the widest. The points of S' are examined in radial order. At the start of the sweep, we set $\ell = \ell_{q_1 p_2}$. The sweep then proceeds in the direction that causes the intersection of $\ell_{p_1 p_2}$ and ℓ to move farther away from p_1. (This may be clockwise or counterclockwise, depending on the relative position of q_1, p_1, p_2.) See Fig. 3 for an example. The numeric labels indicate the order in which the points of S' are visited by the sweep.

As the sweep proceeds, starting from $\ell_{q_1 p_2}$, and up to angle $\angle q_1 p_2 p_1$ from this position, the set $P = S_{\ell m}^{++}(p_1, q_1)$ changes dynamically. We keep track of $CH(P)$, and update it via insertions or deletions, depending on whether the points of S' enter or exit $H_\ell^+(p_1)$.

We are interested in the points p' and q' of P nearest to m and the rotating line ℓ, respectively. Obviously, these points change as the sweep proceeds. For a fixed direction of the rotating line, points p' and q' define the inner boundary of a locally widest corridor if the conditions of Lemma 1 are satisfied. Computing the nearest points can be done in $O(\log n)$ time, as indicated in Observation 1.

We consider three different events during the rotation (see Fig. 4):

(a) ℓ passes through a new point q_2 (initially, $q2 = p2$). In this case we obtain a candidate corridor of type $(21, 21)$.
(b) The nearest point of the current set P to ℓ changes. When this happens ℓ is parallel to one face of $CH(P)$ and we obtain a candidate of type $(21, 12)$.
(c) For the nearest point q' of P to ℓ, the segment from q to q' is perpendicular to ℓ. This gives us a candidate of type $(21, 11)$.

In the example of Fig. 3, only events of type (a) (candidates of type $(21, 21)$) occur. When the sweepline reaches 1, i.e., when $q_2 = 1$, $CH(P) = \langle 3, 5, 8, 9\rangle$, $p' = 9$, and $q' = 3$. Since $\angle 3q_1 1$ is obtuse, these points do not define a locally widest corridor (a small counterclockwise rotation around q_1 and 3, increases the width of the link through q_1). When the sweepline reaches 4, $CH(P) = \langle 1, 2, 5, 8, 9\rangle$, $p' = 9$, $q' = 2$, and we have a locally widest 1-corner corridor of type $(21, 21)$. As this example illustrates, a superset of the locally widest corridors is generated by the process.

The time to perform a radial sort is $O(n \log n)$. By storing $CH(P)$ in a data structure that allows binary search, we can compute all the nearest points of $CH(P)$ to ℓ, as it rotates, in $O(n \log n)$ time. Furthermore, $CH(P)$ can be updated dynamically at an amortized cost of $O(\log n)$ per insertion or deletion [1]. Therefore, the overall time for rotation and processing of events is $O(n \log n)$. Since, there are $O(n^3)$ triples (q_1, p_1, p_2), the best candidate among types $(21, 21)$, $(21, 11)$ and $(21, 12)$ can be computed in $O(n^4 \log n)$ time and $O(n)$ space.

The remaining cases can be solved in a similar way. We briefly describe how to proceed.

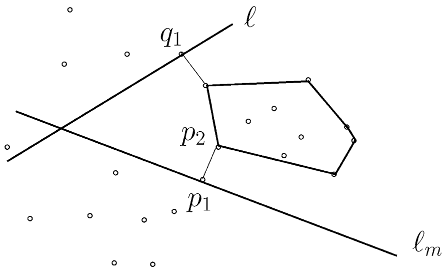

Fig. 5. The case (11, 11).

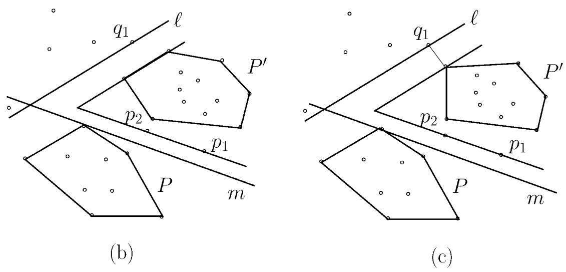

Fig. 6. Cases (12, 12) and (12, 11).

4.2. *Case* (11, 11)

Let p_1, p_2, q_1 be three points of S. Denote by ℓ_m the line through p_1, perpendicular to $\ell_{p_1 p_2}$ and suppose the 1-corner corridor we are looking for has p_1 on one outer leg and q_1 on the other. As before, we will obtain all candidates with a radial sweep of the line through q_1. Indeed, l_m is fixed and we proceed with the rotation of $\ell = \ell_{q_1}$ if p_2 is the point of the current $P = S^{++}_{\ell m}(p_1, q_1)$ closest to the line ℓ_m as in Fig. 5. If so, we rotate ℓ as before but only consider events of type (c). Notice that we can start the rotation with $\ell = \ell_{q_1, p_1}$, updating $CH(P)$ as before. The overall complexity remains the same.

4.3. *Cases* (12, 12), (12, 11)

We proceed as in Section 4.1 but now we keep track of two convex hulls, built on $P = S^{++}_{\ell m}(p_1, q_1)$, and $P' = S^{+-}_{\ell m}(p_1, q_1)$ (see Fig. 6). The point p' of $S^{+-}_{\ell m}(p_1, q_1)$ closest to m determines one leg of our candidate corridor. The other leg is found by rotating the line ℓ anchored at q_1 and stopping at events of types (b) and (c). In fact, an event of type (b) corresponds to case (12, 12) and one of type (c) to the case (12, 11). Once the points p_1, p_2, q_1 are fixed, only $CH(P)$ and $CH(P')$ need to be updated in the sweep. The overall cost is the same as that of Section 4.1.

In summary, we have established the following.

Theorem 1. *Let S be a set of n points. A widest* 1*-corner corridor through S can be computed in* $O(n^4 \log n)$ *time and* $O(n)$ *space.*

5. Widest corridor approximation

Given $0 < \varepsilon < \varepsilon_0$, where ε_0 is a parameter (to be defined later) whose value depends on the input set S, we show how to find a $(1-\varepsilon)$-approximation $\overline{w}$ of the width w^* of a widest empty corridor, i.e., we obtain $(1-\varepsilon)w^* \leqslant \overline{w} \leqslant w^*$. First, we show how to approximate a simple corridor and then adapt our approach to the 1-corner case. Compared to the exact solution, the proposed algorithm is very simple to implement.

Let $\theta_\varepsilon = \sqrt{2\varepsilon}$ and $K = \lfloor \pi/\theta_\varepsilon \rfloor$. Consider the set of equally spaced directions in $[0, \pi]$ given by $\theta_i = \theta_\varepsilon i$, for $i = 0, \ldots, K$. Also set $\theta_{K+1} = \pi$.

Consider a fixed direction θ_i, $i = 0, \ldots, K$, and denote by ϕ_i (resp. ϕ'_i) the direction obtained by rotating θ_i counterclockwise (resp. clockwise) by angle $\pi/2$. Let $P_i = \{p_{i1}, \ldots, p_{in}\}$ be the set of points of S sorted in direction ϕ_i. For $j = 1, \ldots, n-1$, let $w_{i,j}$ be the width of the empty corridor $C_{i,j}$ determined by the two parallel straight lines $\ell_{i,j}$ and $\ell_{i,j+1}$ through $p_{i,j}$ and $p_{i,j+1}$, respectively, in direction θ_i. Denote by $w^*_{i,j}$ the width of the widest empty corridor that can be obtained from $C_{i,j}$ by simultaneously rotating $\ell_{i,j}$ and $\ell_{i,j+1}$ around $p_{i,j}$ and $p_{i,j+1}$, respectively, by an angle less than θ_ε. After such rotation the resulting angle of $\ell_{i,j}$ and $\ell_{i,j+1}$ is $\alpha_{i,j} \in [\theta_i, \theta_{i+1})$. We have the following result,

Lemma 3. $(1-\varepsilon)w^*_{i,j} \leqslant w_{i,j}$, *for* $i = 0, \ldots, K$ *and* $j = 1, \ldots, n-1$.

Proof. Since $\alpha_{i,j} - \theta_i \leqslant \theta_\varepsilon$ and $\cos\theta_\varepsilon \geqslant 1 - \theta_\varepsilon^2/2 = 1 - \varepsilon$, we have,

$$\cos(\alpha_{i,j} - \theta_i) \geqslant 1 - \varepsilon. \tag{1}$$

On the other hand (see Fig. 7),

$$w^*_{i,j} = \overline{p_{i,j}\, p_{i,j+1}} \cos\beta, \tag{2}$$

where β is the angle between the segment $\overline{p_{i,j}\, p_{i,j+1}}$ and the ray from $p_{i,j+1}$ in direction orthogonal to $\alpha_{i,j}$.

Consequently, taking into account (1) and (2), we obtain,

$$\begin{aligned} w_{i,j} &= \overline{p_{i,j}\, p_{i,j+1}} \cos(\beta + \alpha_{i,j} - \theta_i) \\ &\geqslant \overline{p_{i,j}\, p_{i,j+1}} \cos(\beta) \cos(\alpha_{i,j} - \theta_i) \\ &\geqslant w^*_{i,j}(1-\varepsilon). \qquad \square \end{aligned}$$

Let $w_i = \max_j\{w_{i,j}\}$ and $\overline{w} = \max_i\{w_i\}$. Furthermore, let H be the set of lines each of which is either

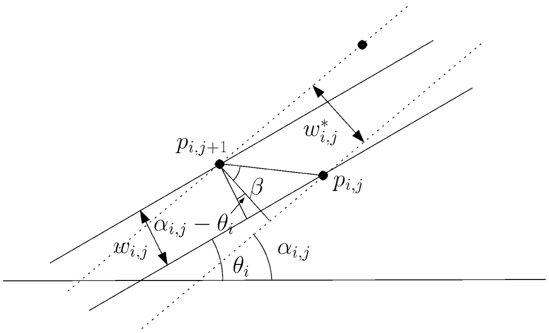

Fig. 7. Corridor approximation.

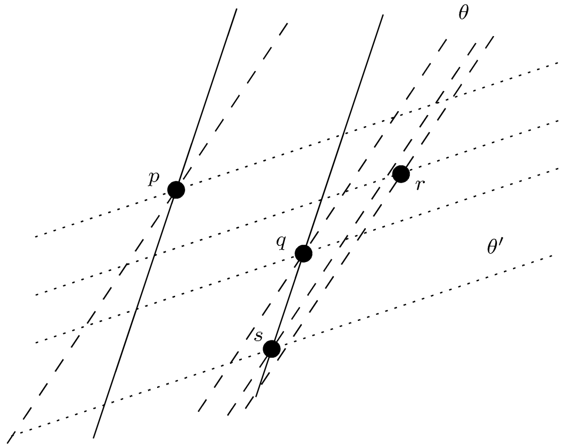

Fig. 8. A set of points and two candidate directions: θ (dashed) produces an optimal solution, while θ' (dotted) does not.

incident on a pair of points from S or orthogonal to the segment joining that pair. Since an optimal corridor C^* has slope in $[\theta_{\bar{\imath}}, \theta_{\bar{\imath}+1})$, for some $\bar{\imath}$, $w^* = \max_{i,j}\{w^*_{i,j}\}$, provided that there are points of S on both legs of C^* that are neighbors in $P_{\bar{\imath}}$. To this end, we require that the set of directions $\{\theta_i\}$ be "sufficiently dense", so that C^* can be produced by rotating some candidate with direction $\theta_{\bar{\imath}}$. This can always be done if $\varepsilon < \varepsilon_0$, where ε_0 is the smallest angle between pairs of lines from H. These ideas are illustrated in Fig. 8. The optimal corridor C^* is given by the solid lines and goes through points p, q, and s. If θ is the nearest direction to C^*, then the optimal solution is found. However, if θ' is the nearest direction, $\{\theta_i\}$ is not dense enough and an optimal corridor is not found as no rotation around p and r can possibly produce C^*. Thus, if $0 < \varepsilon < \varepsilon_0$ then $w^* = w^*_{\bar{\imath},\bar{\jmath}}$, for some $\bar{\jmath}$, and $(1-\varepsilon)w^* = (1-\varepsilon)w^*_{\bar{\imath},\bar{\jmath}} \leqslant w_{\bar{\imath},\bar{\jmath}} \leqslant w_{\bar{\imath}} \leqslant \overline{w} \leqslant w^*$. We have proven the following result,

Lemma 4. *$\overline{w}$ is a $(1-\varepsilon)$-approximation of w^*.*

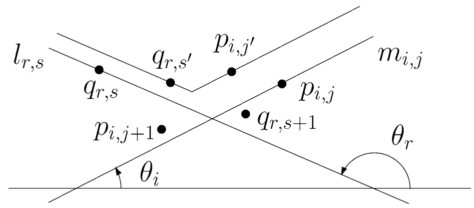

Fig. 9. $C_{i,r}(L_j, L_s)$ 1-corner corridor.

A simple generalization of the approach above allows us to find a $(1-\varepsilon)$-approximation of a widest 1-corner corridor.

Given fixed directions θ_i and θ_r, with $i < j$, there are four types of 1-corner corridors to be considered. We describe in detail the case where the inner boundary is above the outer boundary. The other cases can be handled similarly.

Let $P_i = \langle p_{i1}, \ldots, p_{in}\rangle$ and $Q_r = \langle q_{r1}, \ldots, q_{rn}\rangle$ be the points of S sorted along directions ϕ_i and ϕ'_r (orthogonal to θ_i and θ_r), respectively. Fix $p_{i,j} \in P_i$ and let $m_{i,j}$ be the line through $p_{i,j}$ with direction θ_i. Also, fix $q_{r,s} = p_{i,h} \in Q_r$, $h > j$, and let $l_{r,s}$ be the line through $q_{r,s}$ with direction θ_r. Let $p_{i,j'}$ be the first point of $S^{++}_{l_{r,s}m_{i,j}}$ that follows $p_{i,j}$ in P_i, and let $q_{r,s'}$ be the first point of $S^{++}_{l_{r,s}m_{i,j}}$ that follows $q_{r,s}$ in Q_r. Let $C_{i,r}(L_j, L_s)$ be the 1-corner corridor, with link L_j determined by the line $m_{i,j}$ and the line $m_{i,j'}$ parallel to $m_{i,j}$ through $p_{i,j'}$, and with link L_s determined by the line $l_{r,s}$ and the line $l_{r,s'}$ parallel to $l_{r,s}$ through $q_{r,s'}$ (see Fig. 9).

Let $w_{i,r;j,s}$ be the width of corridor $C_{i,r}(L_j, L_s)$. Proceeding similarly to the case of a simple corridor, let $w^*_{i,r;j,s}$ be the width of the widest empty 1-corner corridor that can be obtained from $C_{i,r}(L_j, L_s)$ by *conceptually* rotating the two links independently as follows: rotate simultaneously the lines $m_{i,j}$ and $m_{i,j'}$ to an angle $\alpha_{i,j} \in [\theta_i, \theta_{i+1})$ around $p_{i,j}$ and $p_{i,j'}$, respectively, and also rotate simultaneously the lines $l_{r,s}$ and $l_{r,s'}$ to an angle $\alpha'_{r,s} \in [\theta_r, \theta_{r+1})$ around $q_{r,s}$ and $q_{r,s'}$, respectively, until a widest corridor is obtained.

Let $w_{i,r} = \max_{j,s}\{w_{i,r;j,s}\}$ and $\overline{w} = \max_{i,r}\{w_{i,r}\}$. Also, let $w^* = \max_{i,r;j,s}\{w^*_{i,r;j,s}\}$. Mimicking the steps in the proofs of Lemmas 3 and 4, it is not difficult to prove that $(1-\varepsilon)w^* \leqslant \overline{w} \leqslant w^*$. In other words,

Lemma 5. *$\overline{w}$ is a $(1-\varepsilon)$-factor approximation of w^*.*

We now analyze an approximation algorithm based on these ideas. Our algorithm computes the width of all 1-corner corridors $C_{i,r}(L_j, L_s)$ described above and reports the maximum $\overline{w}$. In the worst case, we must consider $O(K^2n^2)$ corridors. To do this efficiently we first sort the points of S in the $O(2K)$ directions ϕ_i and ϕ'_i in

$O(Kn\log n)$ time and $O(Kn)$ space. Using the sorted sets, we can compute each corridor and its width in amortized constant time as follows. For fixed θ_i and θ_r, $i < r$, we examine the points of P_i in order. For each point $p_{i,j}$, we examine the points of Q_r, also in order. For each point $q_{r,s} \in Q_r$ in the halfplane bounded by $m_{i,j}$ of direction ϕ_i we proceed as follows: if $p_{i,j}$ is in the halfplane bounded by $l_{r,s}$ of direction ϕ'_r, we find the points $p_{i,j'}$ (resp. $q_{r,s'}$) by moving forward along the sorted list P_i (resp. Q_r) to the first occurrence inside $S^{++}_{l_{r,s}m_{i,j}}$. (Notice that possibly $p_{i,j'} = q_{r,s'}$.) Testing if a point is inside a halfplane or inside the wedge $S^{++}_{l_{r,s}m_{i,j}}$ is done in constant time. Furthermore, once $p_{i,j}$, $p_{i,j'}$, $q_{r,s}$, $q_{r,s'}$, are known, the cost of $C_{i,r}(L_j, L_s)$ is also computed in constant time. Notice that the point of Q_r that determines the additional leg of the link containing $q_{r,s+1}$ (the successor of $q_{r,s}$) is either identical to $q_{r,s'}$ or follows it in Q_r. In other words, since for each fixed $p_{i,j}$ the processing of lists P_i and Q_s always moves forward, it takes $O(n)$ time to compute all corridors with links whose directions are θ_i and θ_r and with a leg through $p_{i,j}$. Consequently, we can find $\overline{w}$ in

$$O(Kn\log n + K^2n^2) = O\left(\frac{n\log n}{\varepsilon^{1/2}} + \frac{n^2}{\varepsilon}\right)$$

time and $O(n/\varepsilon^{1/2})$ space. Finally, we point out that ε_0 is only needed to guarantee the correctness and its value is not needed by the algorithm. If the user wants to establish a range of possible values for ε, ε_0 can be computed in $O(n^2\log n)$ time by sorting H by slope and finding the two neighbor lines in the sorted list with smallest difference in slope. In summary, we have proven the following result:

Theorem 2. *An $(1-\varepsilon)$-approximation of a widest empty 1-corner corridor through S can be computed in $O((n\log n)/\varepsilon^{1/2} + n^2/\varepsilon)$ time and $O(n/\varepsilon^{1/2})$ space, for $\varepsilon < \varepsilon_0$.*

Acknowledgements

The authors would like to thank the anonymous referees who pointed out how to simplify the description of the algorithms and suggested numerous improvements. This problem was posed during the *First Spanish Workshop on Geometric Optimization*, held in the Summer of 2002. The first author was supported by grant BMF2003-04062. The second author was supported by the National Science Foundation under grant DMS-0107628. The third author was supported by grant TIN2004-08065-C02-02.

References

[1] G.S. Brodal, R. Jacob, Dynamic planar convex hull, in: Proc. 43rd Annual IEEE Symp. on Foundations of Computer Science, 2002, pp. 617–626.

[2] S.W. Cheng, Widest empty L-shaped corridor, Inform. Process. Lett. 58 (1996) 277–283.

[3] J.M. Díaz-Báñez, F. Hurtado, Computing obnoxious 1-corner polygonal chains, Comput. Oper. Res. 33 (2006) 1117–1128.

[4] J.M. Díaz-Báñez, F. Hurtado, H. Meijer, D. Rappaport, J.A. Sellarès, The largest empty annulus problem, Internat. J. Comput. Geom. Appl. 13 (2003) 317–325.

[5] H. Edelsbrunner, E.P. Müecke, Simulation of simplicity, ACM Trans. Graphics 9 (1990) 66–104.

[6] M. Houle, A. Maciel, Finding the widest empty corridor through a set of points, TR SOCS-88.11, Department of Computer Science, McGill University, Montreal, Canada, 1988.

[7] R. Janardan, F.P. Preparata, Widest-corridor problems, Nordic J. Comput. 1 (1994) 231–245.

[8] A. Naamad, W.L. Hsu, D.T. Lee, On the maximum empty rectangle problem, Discrete Appl. Math. 8 (1984) 267–277.

[9] S.C. Nandy, B.B. Bhattacharya, On finding an empty staircase polygon of largest area (width) in a planar point set, Comput. Geom. Theory Appl. 26 (2003) 143–171.

[10] G.T. Toussaint, Computing largest empty circles with location constraints, Internat. J. Comput. Inform. Sci. 12 (1983) 347–358.