Finding a widest empty 1-corner corridor

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

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

Abstract

Given a set of n points in the plane, we consider the problem of computing a widest empty 1-corner corridor. We star giving a characterization of the 1-corner corridors that we call *locally widest*. Our approach to finding a widest empty 1-corner corridor consists of identifying a set of 1-corner corridors locally widest, that is guaranteed to contain a solution. We describe an algorithm that solves the problem in $O(n^4 \log n)$ time and $O(n)$ space.

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 $r(L) = p + \boldsymbol{v}t$ and $r'(L) = p' + \boldsymbol{v}t$, and a 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* $\alpha(C), 0 < \alpha(C) \leq \pi$, 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. Note that 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 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 [Che96] (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 have applications in robot manipulation and spatial planning.

The problem of computing a widest empty corridor can be solved in $O(n^2)$ time and $O(n)$ space [HM88,JP94]. 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 [Che96]. In this paper we allow $\alpha(C)$ to assume arbitrary values and describe an algorithm to compute a widest empty 1-corridor in $O(n^4 \log n)$ time and $O(n)$ space. It is clear that the solution to this problem might be not unique. In this paper we just look for one optimal corridor.

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

Due to space constraint, in this extend abstract most proofs have been omitted.

* Corresponding author

Email addresses: dbanez@us.es (J.M. Díaz-Báñez), mlopez@cs.du.edu (M. A. López), sellares@ima.udg.es (J.A. Sellarès).

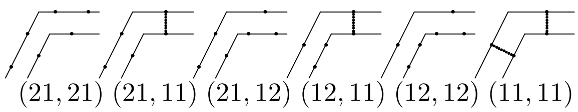

Fig. 1. Six types of locally widest corridors. The interior segments are perpendicular to the incident boundary legs.

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 $\alpha(C) < \pi$.

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.

We say that a 1-corner corridor is *locally widest* if each leg contains at least one point of S and it is not possible to increase the width of either link by performing rotations of the legs around the points from S incident on the leg boundaries.

Lemma 1 *Each link L of a locally widest corridor 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 p' of S that lies on the inner leg $r'(L)$, such that both $\angle p'p_1p_2$ and $\angle p'p_2p_1$ are acute.

12: There are points p'_1 and p'_2 of S that lie on the inner leg $r'(L)$ and a point p of S that lies on the outer leg $r(L)$, such that both $\angle pp'_1p'_2$ and $\angle pp'_2p'_1$ are acute.

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

From the characterization given in Lemma 1 it results 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 Figure 1.

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

Our approach to finding a widest empty 1-corner corridor consists of identifying a set $\mathcal{C}$ of locally widest 1-corner corridors that, by Lemma 2, is guaranteed to contain a solution. Our algorithm operates by systematically generating $\mathcal{C}$. Since there are members of $\mathcal{C}$ with six points on the boundary, a brute force algorithm would run in $O(n^7)$ time. Instead, our algorithm generates one boundary for each link, and computes the remaining boundaries by translating and appropriately adjusting a copy of a boundary ray until a point of S is encountered. This allows us to compute the optimal 1-corner corridor in $O(n^4 \log n)$ time.

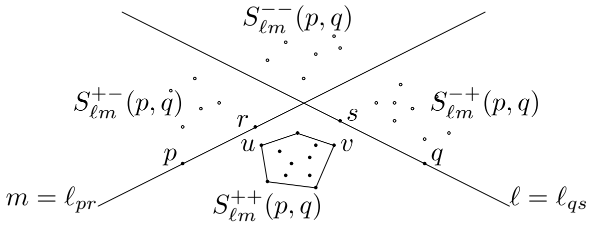

Fig. 2. Partition of the plane by boundary legs. The points u and v of $S^{++}_{\ell m}(p,q)$, closest to the outer boundary legs, help define the links of the corridor.

3. Preliminaries

For any two points p and q we denote by ℓ_{pq} the line through p and q and for any point t and line ℓ we denote by $H^+_\ell(t)$ (resp. $H^-_\ell(t)$) the open halfplane bounded by ℓ that contains (resp. does not contain) t. 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 $H^+_\ell(p) \cap H^+_m(q)$. Again, when $\ell = \ell_{qs}$ and $m = \ell_{pr}$, we may write $S^{++}_{qspr}(p,q)$ to denote the same set of points. The other three regions determined by ℓ and m are labelled $S^{--}_{\ell m}(p,q)$, $S^{-+}_{\ell m}(p,q)$, and $S^{+-}_{\ell m}(p,q)$, as illustrated in Figure 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 take advantage of 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, the point of P closest to ℓ can be computed in $O(\log m)$ time.*

In the example of Figure 2, if P is the set of points in $S^{++}_{\ell m}(p,q)$, then u and v are the points of P closest to ℓ_{pr} and ℓ_{qs}, respectively. Points u, p and r (resp. v, q and s), 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 [EM90].

Case (21,21).

Consider first an optimal corridor of type $(21, 21)$. Such a corridor contains points (p_1, p_2) on one outer leg and points (q_1, q_2) on the other. We do not discard the 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 will compute all candidate corridors of type $(21, 21)$ for each triple (q_1, p_1, p_2) of points from S.

For each point $q_1 \in S$, we start by computing the radial ordering of $S \setminus \{q_1\}$ as a line ℓ through q_1 rotates around q_1. The initial orientation of the rotating line is arbitrary and rotation angles in the range $[0, \pi)$ suffice so that each input point is visited exactly once by one of the two rays making up the rotating line. In practice, and 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.

For each ordered pair (p_1, p_2) of points from $S \setminus \{q_1\}$, let $m = \ell_{p_1p_2}$ and $S' = S \cap H_m^+(q_1)$. The idea is to consider all corridors of type $(21, 21)$ 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 and each allowed to take on the role of q_2, i.e., the other point on the same outer leg as q_1. Initially, we set $q_2 = p_2$, as $\ell_{q_1p_2}$ is the position of the rotating line ℓ at the start of the sweep. The sweep then proceeds in the direction that causes the intersection of $\ell_{p_1p_2}$ and ℓ to move farther away from p_1. (This may be clockwise or counter-clockwise, depending on the relative position of q_1, p_1, p_2.) See Figure 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 $q_2 = p_2$, and up to angle $\angle q_1p_2p_1$ 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 $H_\ell^+(p_1)$. For each point visited, we compute a tentative inner boundary by finding

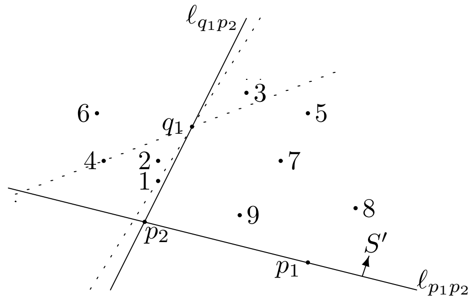

Fig. 3. Computing corridors of type $(21, 21)$ based on points q_1, p_1 and p_2. Solid line $\ell_{q_1p_2}$ defines the start of the sweep. Dashed linesillustrate different events of the sweep. Numeric labels indicate the order in which the points of S' are visited by the sweep.

the points p' and q' of P nearest to $m = \ell_{p_1p_2}$ and $\ell = \ell_{q_1q_2}$, respectively. This can be done in $O(\log n)$ time, as indicated in Observation 1. Points p' and q' define the inner boundary of a locally widest corridor if $\angle p'p_1, p_2$, $\angle p'p_2p_1$, $\angle q'q_1q_2$, and $\angle q'q_2q_1$ are all acute. In the example of Figure 3, 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 $\angle 3q_11$ is obtuse, these points do not define a locally widest corridor (a small counter-clockwise rotation around q_1 and 3, increases the width of the link through q_1). When the sweepline reaches 4, $CH(P) = \langle 1, 2, 5, 8, 9\rangle$, $p' = 9$, and $q' = 2$ and we have a valid 1-corner corridor of type $(21, 21)$.

The time to perform a radial sort is $O(n \log n)$. Since $CH(P)$ can be updated dynamically at an amortized cost of $O(\log n)$ per insertion or deletion [BJ02], the cost of one radial sweep is $O(n \log n)$. Since, there are $O(n^3)$ triples (q_1, p_1, p_2), the best candidate of type $(21, 21)$ can be found in $O(n^4 \log n)$ time and $O(n)$ space.

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

Case (21,11).

This case is handled by a simple modification to the ideas discussed above. As before, p_1 and p_2 lie on one outer leg, and q_1 lies on the other. Let ℓ denote the rotating line, and $\ell^\perp$ the line through q_1, perpendicular to ℓ. We handle an event every time ℓ goes through a point of S'. Unlike the previous case, these events serve only the purpose of keeping $CH(P)$ up to date and no candidate corridors are generated. Additionally, we handle an event every time $\ell^\perp$ goes through a point q. When this happens, the points p' and q' of P closest to $\ell_{p_1p_2}$ and ℓ

are computed, and a corridor of type (12,11) is generated if $q' = q$ and angles $\angle p'p_1p_2$ and $\angle p'p_2p_1$ are both acute. The complexity of both time and space remains the same as before.

Case (11,11).

This case is similar to $(21, 11)$ except for the interpretation of p_1 and p_2. Without lost of generality we could suppose that points p_1 and q_1 lie on the outer legs of the 1-corner corridor. For each pair p_1 and p_2 of points from S, if p_2 is the point of S' closest to the line m orthogonal to $\overline{p_1p_2}$ through p_1 then a sweep similar to that used for $(21, 11)$ is performed. The complexity remains the same.

Case (21,12).

The rotating line starts out parallel to $m = \ell_{p_1p_2}$. We keep track of two convex hulls, one built on $P = S^{++}_{\ell m}(p_1, q_1)$, and another built on $P' = S^{-+}_{\ell m}(p_1, q_1)$. For every point q_2 of S visited by the rotating ℓ, we find the point p' of P closest to m, and the point q' of P' closest to ℓ. The corridor is valid if the line parallel to ℓ through q' intersects with line m in such a way that points p_1, p_2 lie on the outer leg determined by m, and if angles $\angle p'p_1p_2$, $\angle p'p_2p_1$, $\angle q'q_1q_2$ and $\angle q'q_2q_1$ are all acute. Again the complexity remains the same.

Case (12,12).

We keep track of three convex hulls, built on $P = S^{++}_{\ell m}(p_1, q_1)$, $P' = S^{-+}_{\ell m}(p_1, q_1)$ and $P'' = S^{--}_{\ell m}(p_1, q_1)$. Find the point p' of $S^{+-}_{\ell m}(p_1, q_1)$ closest to m and the point q' of $S^{-+}_{\ell m}(p_1, q_1)$ closest to ℓ. Let m' be the line through p' parallel to m, ℓ' be the line through q' parallel to ℓ and denote $R = H^{+}_{m'}(p_1) \cap H^{+}_{\ell'}(q_1)$. If no point of $S^{--}_{\ell m}(p_1, q_1)$ lies inside R (condition that can be tested in $O(\log n)$ time by checking that ℓ' and m' do not intersect $CH(P'')$ and that any vertex v of $CH(P'')$ is outside R) then we are done. Otherwise we want to find the points of $S^{--}_{\ell m}(p_1, q_1)$ interiors to R closest to m and to ℓ, respectively. Find the points p'' and q'' of $S^{--}_{\ell m}(p_1, q_1)$ closest to m and ℓ, respectively. If q'' (p'') lies outside the region R, then find the edge s (t) of $CH(P'')$ intersected by ℓ' (m') that is interior in part to R and, abusing of language, denote by q'' (p'') the endpoint of s (t) that lies inside R. Observe that point q'' (p'') can be computed in $O(\log n)$ time. Compute a smaller corridor from the points p', p'', q', q'' found. If we denote p^* and q^* the points that define such a corridor, then the corridor is valid if the line parallel to ℓ through q^* intersects with line m in such a way that points p_1, p_2 lie on the outer leg determined by m, and if angles $\angle p^*p_1p_2$, $\angle p^*p_2p_1$, $\angle q^*q_1q_2$ and $\angle q^*q_2q_1$ are all acute. The best candidate of type (12,12) can be found in $O(n^4 \log n)$ time and $O(n)$ space.

Case (12,11).

This case may be handled by combining the ideas used to solve cases (21,11) and (12,12). Now $\overline{q_1q^*}$ is orthogonal to ℓ. Again the complexity remains the same.

In summary, we have established the following.

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

5. Future work

We are presently working on other variants of the problem, which include finding a widest empty k-dense 1-corner corridor, dynamic updates, and approximation algorithms.

6. Acknowledgments

This problem was posed and partially solved during the *First Spanish Workshop on Geometric Optimization*, held in the Summer of 2002. The first author was supported in part by grant BFM2000-1052.The second author was supported in part by the National Science Foundation under grant DMS-0107628. The The third author was supported in part by grants TIC2001-2392-C03-01 and Gen. Catalunya DURSI-2001SGR-00296.

References

[BJ02] G. S. Brodal and R. Jacob, "Dynamic planar convex hull", *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, pp 617-626, 2002.

[Che96] Siu-Wing Cheng, "Widest empty L-shaped corridor", *Information Processing Letters*, 58, pp 277-283, 1996.

[EM90] H. Edelsbrunner, E. P. Müecke, "Simulation of Simplicity", *ACM Transactions on Graphics*, 9, pp 66-104, 1990.

[HM88] M. Houle, A. Maciel, "Finding the widest empty corridor through a set of points", In G.T. Toussaint, editor *Snapshots of computational and discrete geometry*, pp 210-213 , 1988.

[JP94] R. Janardan, F.P. Preparata, "Widest-corridor problems", *Nordic Journal of Computing*, 1, pp 231-245, 1994.