[Va53] Valentine, F. A., “Minimal sets of visibility,” *Proc. American Mathematical Society*, vol. 4, 1953, pp. 917-921.

[Va70] Valentine, F. A., “Visible shorelines,” *American Mathematical Monthly*, vol. 77, 1970. pp. 146-152.

[Wa76] Wagner, N. R., “The sofa problem,” *American Mathematical Monthly*, vol. 83, 1976, pp. 188-189.

[We88] Wenger, R., “Stabbing and separation,” Ph.D. thesis, School of Computer Science, McGill University, February 1988.

plane," *Geometriae Dedicata,* vol. 9, 1980, pp.461-465.

[Me83] Megiddo, N., "Linear-time algorithms for linear programming in $\mathbf{R}^3$ and related problems," *SIAM Journal on Computing*, Vol. 12, 1983, pp. 759-776.

[Ni81] Niven, I., *Maxima and Minima Without Calculus*, Published by the Mathematical Association of America, 1981.

[OAMB86] O'Rourke, J., Aggarwal, A., Maddila, S. and Baldwin, M., "An optimal algorithm for finding minimal enclosing triangles," *Journal of Algorithms*, vol. 7, 1986, pp. 258-269.

[O'R87] O'Rourke, J., *Art Gallery Theorems and Algorithms*, Oxford University Press, 1987.

[O'R81] O'Rourke, J., "An on-line algorithm for fitting straight lines between data ranges," *Communications of the ACM*, vol. 24, No. 9, September 1981, pp.574-578.

[OV81] Overmars, M. and van Leeuwen, H., "Maintenance of configurations in the plane," *Journal of Computer & System Sciences*, vol. 23, 1981, pp. 166-204.

[Ro88] Robert, J.-M., "Stabbing hyperspheres by a hyperplane," in *Snapshots of Computational and Discrete Geometry*, G. T. Toussaint, ed., Tech. Rep. SOCS-88.11, Computational Geometry Lab., McGill University, June 1988, pp. 181-188.

[Ro85] Rohnert, H., "Shortest paths in the plane with convex polygonal obstacles," Tech. Rept. A85/06, University of Saarbrucken, 1985.

[SH76] Shamos, M. I. and Hoey, D., "Geometric intersection problems," *Seventeenth Annual IEEE Symposium on the Foundations of Computer Science,* October 1976, pp. 208-215.

[Sha89] Sharir, M., "The shortest watchtower and related problems for polyhedral terrains," *Information Processing Letters*, in press.

[SS86] Sack, J.-R. and Suri, S., "An optimal algorithm for detecting weak visibility of a polygon," Tech. Rept. SCS-TR-114, Carleton University, Ottawa, Dec. 1986.

[ST88] Shermer, T. and Toussaint, G. T., "Characterizations of convex and star-shaped polygons," in *Snapshots of Computational and Discrete Geometry,* G. Toussaint, ed., Rept. SOCS-88.11, School of Computer Science, McGill Univ., June 1988.

[Te88] Teichman, M., "Shoving a table into a corner," in *Snapshots of Computational and Discrete Geometry,* G. T. Toussaint, ed., Tech. Report SOCS-88.11, Computational Geometry Laboratory, McGill University, June 1988, pp. 99-118.

[To83] Toussaint, G. T., "Solving geometric problems with the rotating calipers," *Proc. MELECON'83,* Athens, Greece, 1983.

[Ed85b] Edelsbrunner, H., “Computing the extreme distances between two convex polygons,” *Journal of Algorithms*, vol. 6, 1985, pp. 213-224.

[EOW81] Edelsbrunner, H., Overmars, M. H., and Wood, D., “Graphics in flatland: a case study,” Tech. Report F79, Technical University of Graz, 1981.

[EMPRWW82] Edelsbrunner, H., Maurer, H. A., Preparata, F. P., Rosenberg, A. L., Welzl, E., and Wood, D., “Stabbing line segments,” *BIT*, vol. 22., 1982, pp.274-281.

[ET85] ElGindy, H. and Toussaint, G. T., “Efficient algorithms for inserting and deleting edges from triangulations,” *Proc. Intl. Conf. on Foundations of Data Organization,* Kyoto, Japan, May 1985.

[Gr58] Grunbaum, B., “On common transversals,” *Arch. Math.,* vol. 9, 1958, pp. 465-469.

[He89] Hershberger, J., “Finding the upper envelope of n line segments in O(n log n) time,” *Information Processing Letters*, vol. 33, 1989, pp. 169-174.

[HM88] Houle, M. and Maciel, A., “Finding the widest empty corridor through a set of points,” in *Snapshots of Computational and Discrete Geometry,* G. T. Toussaint, ed., Tech. Report SOCS-88.11, Computational Geometry Laboratory, McGill University, June 1988, pp. 201-214.

[Ho88] Houle, M., “A measure of separability for point sets,” in *Snapshots of Computational and Discrete Geometry,* G. T. Toussaint, ed., Tech. Report SOCS-88.11, Computational Geometry Laboratory, McGill University, June 1988, pp. 21-36.

[HV49] Horn, A. and Valentine, F. A., “Some properties of L-sets in the plane,” *Duke Mathematics Journal,* vol.16, 1949, pp.131-140.

[Ka61] Kazarinoff, N. D., *Geometric Inequalities*, Published by the Mathematical Association of America, 1961.

[Ke88] Ke, Y., “Detecting the weak visibility of a simple polygon and related problems,” The Johns Hopkins University, manuscript, March 1988.

[KS86] Kirkpatrick, D. G. and Seidel, R., “The ultimate planar convex hull algorithm?” *SIAM Journal on Computing,* vol. 15, No. 1, February 1986, pp. 287-299.

[KL85] Klee, V. and Laskowski, M. C., “Finding the smallest triangles containing a given convex polygon,” *Journal of Algorithms,* vol. 6, 1985, pp. 359-375.

[La59] Lange, L. H., “Cutting certain minimal corners,” *Mathematics Magazine,* vol. 32, 1959, pp. 157-160.

[Le80] Lewis, T., “Two counterexamples concerning transversals for convex subsets of the

[AW88] Avis, D. and Wenger, R., "Polyhedral line transversals in space," *Discrete and Computational Geometry*, vol. 3, No. 3, 1988, pp. 257-266.

[BET91] Bhattacharya, B. K., Egyed, P., & Toussaint, G. T., "Computing the wingspan of a butterfly," manuscript in preparation.

[BKT89] Bhattacharya, B. K., Kirkpatrick, D. G., & Toussaint, G. T., "Determining sector visibility of a polygon," *Proc. 5th ACM Symposium on Computational Geometry*, Saarbruchen, 1989, pp.247-254.

[BL83] Bajaj, C. and Li, M., "On the duality of intersection and closest points," *Proc. 21st Allerton Conference*, 1983, pp.459-461.

[BV76] Buchman, E. and Valentine, F. A., "External visibility," *Pacific Journal of Mathematics*, vol. 64, 1976, pp. 333-340.

[CD80] Chazelle, B. M. and Dobkin, D. P., "Detection is easier than computation," *Proc. 12th Annual ACM Symposium on the Theory of Computing*, 1980, pp. 146-153.

[CE88] Chazelle, B. M. and Edelsbrunner, H., "An optimal algorithm for intersecting line segments in the plane," *29th Annual Symposium on Foundations of Computer Science*, October 1988, pp. 590-600.

[CL71] Chakerian, G. D. and Lange, L. H., "Geometric extremum problems," *Mathematics Magazine*, vol. 44, 1971, pp. 57-69.

[CL85] Ching, Y. T. and Lee, D. T., "Finding the diameter of a set of lines," *Pattern Recognition*, vol. 18, 1985, pp. 249-255.

[CY84] Chang, J. S. and Yap, C. K., "A polynomial solution for potato-peeling and other polygon inclusion and enclosure problems," *Proc. Foundations of Computer Science*, West Palm Beach, Fla., 1984, pp. 408-416.

[DA84] DePano, N. A. A., and Aggarwal, A., "Finding restricted k-envelopes for convex polygons," *Proc. 22nd Allerton Conference*, Urbana, Ill., 1984, pp. 81-90.

[DK83] Dobkin, D. P. and Kirkpatrick, D. G., "Fast detection of polyhedral intersection," *Theoretical Computer Science*, vol. 27, 1983, pp. 241-253.

[DT90] Devroye, L. and Toussaint, G. T., "Convex hulls for random lines," Tech. Rept. SOCS-90.11, School of Computer Science, McGill University, May 1990.

[Dy84] Dyer, M. E., "Linear time algorithms for two- and three-variable linear programs," *SIAM Journal of Computing*, Vol. 13, No. 1, February 1984, pp. 31-45.

[Ed85] Edelsbrunner, H., "Finding transversals for sets of simple geometric figures," *Theoretical Computer Science*, vol. 35, 1985, pp.55-69.

tion **L** of lines in R^3 there are many possible ways of defining the girth of **L**. One possibility is the circumference of the smallest disc that intersects every member of **L**. It is an open problem to determine the complexity of computing such a disc.

Finally we mention the *sofa* problem [Wa76]. The traditional sofa problem asks for, given a right-angled corridor, what is the rigid object (sofa) of largest area that can be moved around the corner. This problem remains unsolved although upper and lower bounds are known. In case the sofa is a line segment (ladder) the solution is given by Niven [Ni81]. It is not difficult to see that the longest ladder that can be passed around the corner is equivalent to the shortest line segment in the corridor that will connect the outer walls while remaining in contact with point q, the vertex of the inner walls. Therefore lemma 2.1 generalizes the result of Niven [Ni81] to handle corridors which need not be right-angled nor have parallel walls.

7. Acknowledgments

The research reported here was supported by the Natural Sciences & Engineering Research Council of Canada, FCAR in Quebec, and the British Columbia Advanced Systems Institute. The research was carried out while the second author was an Advanced Systems Institute Fellow at Simon Fraser University during the fall of 1988. The authors are grateful to David Dobkin, David Kirkpatrick, Victor Klee and Slawomir Pilarski for fruitful discussions on this topic.

8. References

[AB87] Atallah, M. J. and Bajaj, C., "Efficient algorithms for common transversals," *Information Processing Letters*, vol. 25, May 1987, pp.87-91.

[AD90] Avis, D. and Doskas, M., "Algorithms for high dimensional stabbing problems," *Discrete and Applied Mathematics*, to appear in 1990.

[AGT86] Avis, D., Gum, T., and Toussaint, G. T., "Visibility between two edges of a simple polygon," *The Visual Computer*, vol. 2, 1986, pp. 342-357.

[AHU83] Aho, A. V., Hopcroft, J. E., & Ullman, J. D., *Data Structures and Algorithms*, Addison-Wesley, 1983.

[At86] Atallah, M. J., "Computing the convex hull of line intersections," *Journal of Algorithms*, vol. 7, 1986, pp. 285-288.

[AT81] Avis, A. and Toussaint, G. T., "An optimal algorithm for determining the visibility of a polygon from an edge," *IEEE Transactions on Computers*, vol. C-30, No. 12, December 1981, pp.910-914.

[AW87] Avis, D. and Wenger, R., "Algorithms for line stabbers in space," *Proc. 3rd ACM Symposium on Computational Geometry*, 1987, pp.300-307.

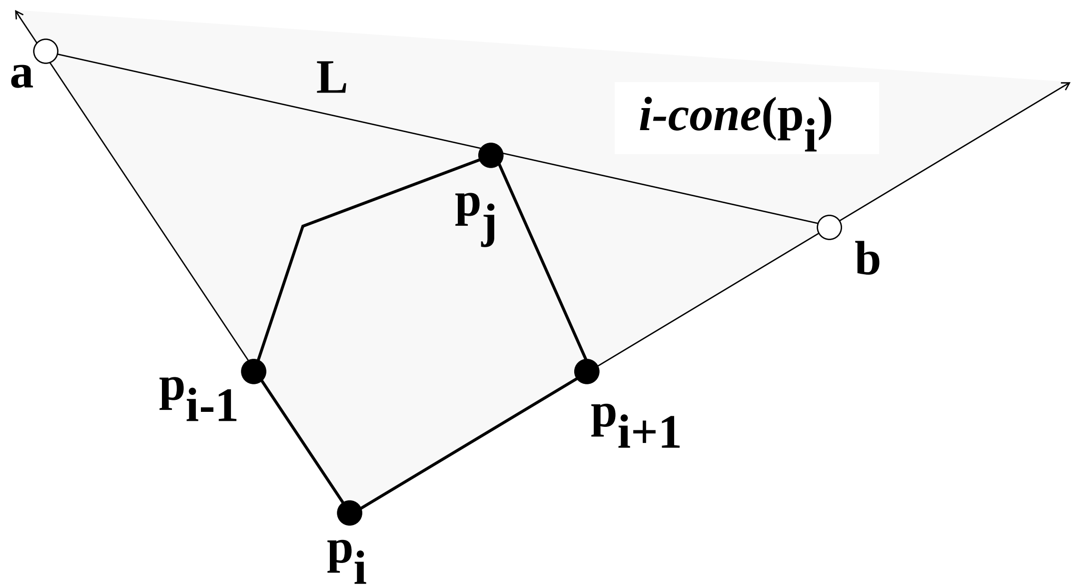

Fig. 5.1: Illustrating the internal cone of support of vertex p_i of a convex polygon P.

them. The algorithm in [AGT86] for determining partial visibility between two edges in P computes the "internal convex hulls" between the end points of the edges to create an "hourglass" polygon if in fact the edges are partially visible thus leading to a situation similar to that encountered in section 2 where we have two instances of the geometric minimization. Therefore, combining the results of section 2 with those in [AGT86] we obtain the following.

Theorem 5.2: Given two edges $\mathbf{e}_1$ and $\mathbf{e}_2$ in a simple n-vertex polygon P, the *shortest* line-of-sight between them, if one exists, can be computed in O(n) time and space.

6. Concluding Remarks

In this paper it was shown that given a set of lines $\mathbf{L} = \{L_1, L_2,...,L_n\}$ the *girth* of **L**, or the shortest line segment that intersects every member of **L**, can be found in $O(n \log^2 n)$ time and O(n) space. This was established by proving that the shortest transversal of a set of lines **L** is identical to the shortest transversal of a set of line-segments **S** which are obtained by intersecting each given line in **L** with the convex hull of the intersection points determined by these lines. An $O(n \log^2 n)$ time algorithm for arbitrary line segments is subsequently used on **S**. Note however that **S** is not an arbitrary set of line segments as it exhibits the property that all the end points of all the line segments lie on the boundary of a convex polygon. It may be the case that exploiting this information will yield an O(n log n) time algorithm for computing the girth of an arrangement. Given a collec-

Let L*(P) be the shortest line segment from which P is weakly externally visible.

Lemma 5.3: L*(P) has one of its end points on one of the bounding rays of $W_{in}(e_i)$ and the other on the other bounding ray for some value of i.

Lemma 5.4: L*(P) is tangent to P.

Definition: Let p_i be a vertex of P. The *internal cone of support* at p_i, denoted by *i-cone*(p_i), is the wedge determined by $HL(p_{i-1},p_i) \cap HL(p_i,p_{i+1})$.

Lemma 5.5: A line segment L=[a,b] lying in *i-cone*(p_i), tangent to P, from which P is weakly visible must be tangent to a vertex p_j such that p_j is an *antipodal* vertex of p_i.

Proof: If L does not intersect *ray*(p_i,p_{i-1}) then edge [p_{i-1},p_i] of P is not visible from L. Similarly, if L does not intersect *ray*(p_i,p_{i+1}) then edge [p_i,p_{i+1}] of P is not visible from L. Therefore L must have one endpoint on *ray*(p_i,p_{i-1}) and the other on *ray*(p_i,p_{i+1}). Now assume that p_j is not an *antipodal* vertex of p_i. Let $p_{k(i-1,i)}$ denote the vertex of P determined by a tangent line parallel to *ray*(p_i,p_{i-1}) and antipodal to p_{i-1} and p_i. Similarly, Let $p_{k(i,i+1)}$ denote the vertex of P determined by a tangent line parallel to *ray*(p_i,p_{i+1}) and antipodal to p_{i+1} and p_i. Clearly, as we rotate a line of support in a counterclockwise manner starting at $p_{k(i-1,i)}$ and ending at $p_{k(i,i+1)}$ we visit the polygonal chain $C[p_{k(i-1,i)}, p_{k(i-1,i)+1},...,p_{k(i,i+1)-1}, p_{k(i,i+1)}]$ which has the property that all its vertices are antipodal to p_i. Furthermore if p_j is not antipodal to p_i then L must have an unoriented direction that lies in the wedge determined by the internal angle of P at vertex p_i. This implies that one endpoint of L must lie in *int*(*i-cone*(p_i)), which is a contradiction. Q.E.D.

Lemma 5.6: The antipodal chains of two consecutive cones intersect only at their end points.

The above lemmas combined with the results of section 2 lead straightforwardly to the following theorem the details of which are left to the reader.

Theorem 5.1: Given a convex n-gon P, the shortest line segment L*(P) from which P is weakly externally visible can be found in O(n) time.

5.2. Shortest Lines of Sight

One of the most recurring themes in many computer applications such as graphics, automated cartography, robotics and image processing is the notion of visibility relations between elements such as vertices or edges in a simple polygon. One such class of problems is concerned with computing *edge-to-edge* visibility relations. Given two edges $\mathbf{e}_1$ and $\mathbf{e}_2$ of a simple n-vertex polygon P, there exist four natural types of visibility all of which can be determined in O(n) time [AGT86]. A *line-of-sight* between $\mathbf{e}_1$ and $\mathbf{e}_2$ is a line segment [*a*,*b*] such that $a \in \mathbf{e}_1$ and $b \in \mathbf{e}_2$ and $[a,b] \in$ P. Two edges $\mathbf{e}_1$ and $\mathbf{e}_2$ of P are said to be *partially-visible* if they admit a *line-of-sight* between

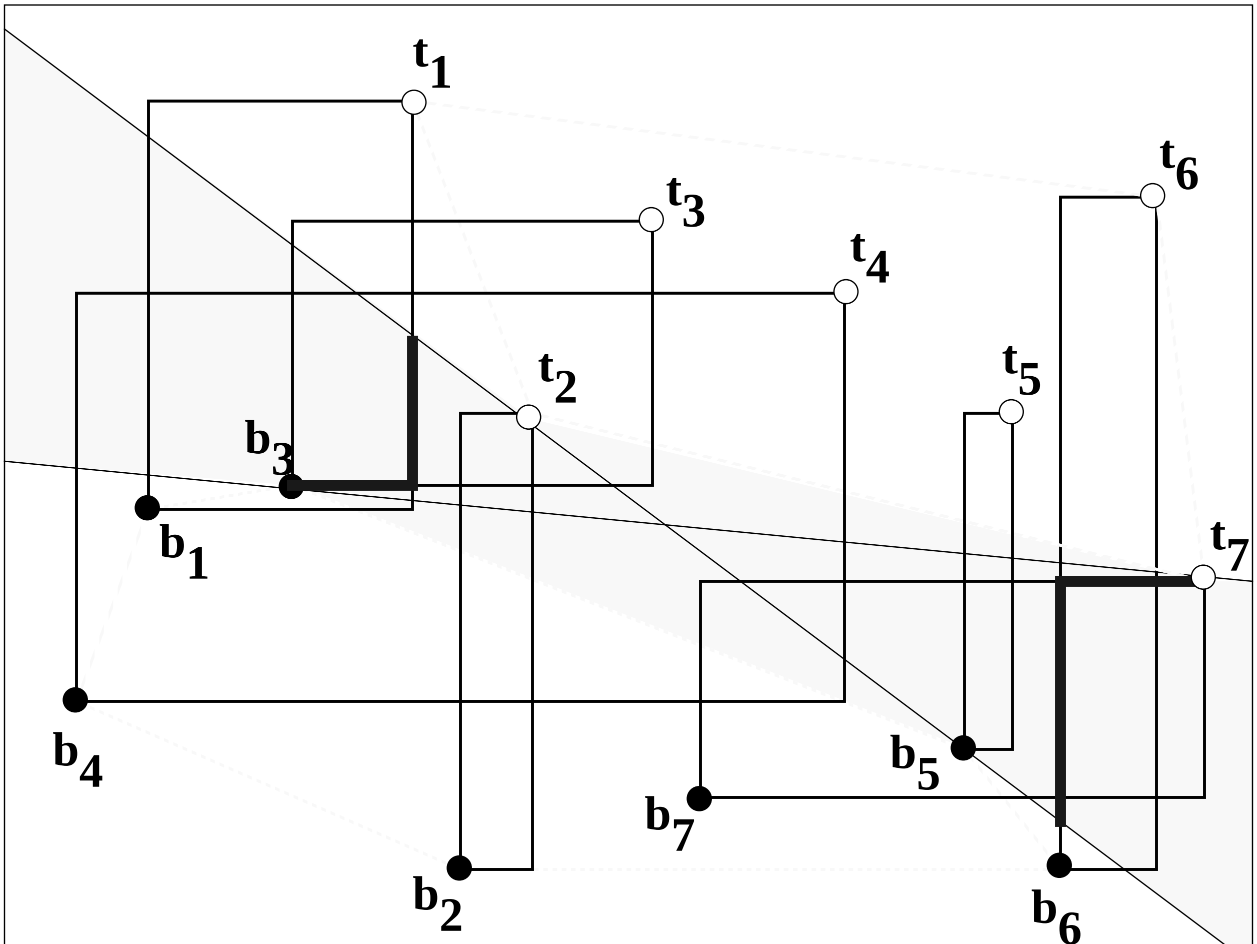

Fig. 4.6   Illustrating one of the two possible regions (shaded) in which all line transversals of a set of isothetic rectangles must lie. The sets $\boldsymbol{F}_t$ and $\boldsymbol{F}_b$ are shown in white and black filled circles, respectively. Heavy lines indicate the left and right envelopes.

If a half-plane contains the interior of P it will be referred to as an *interior* half plane. Denote the infinite half-ray starting at a point x and traversing a second point y by *ray*(x,y). Let L=[a,b] (also just plain L) denote a line segment with end points a,b. For a vertex p_i of P the lines through e_{i-1} and e_i partition the plane into four wedges. Let $W_{in}(p_i)$ be the wedge containing P. We begin by presenting several easy lemmas that we state without proof.

Lemma 5.1: P is weakly externally visible from L[a,b] if, and only if, for i=1,2,...,n we have that $HR(e_i) \cap L[a,b] \neq \emptyset$.

Lemma 5.2: P is weakly externally visible from L[a,b] if, and only if, there exists a tangent ray of support to P from a and b, ray(a,P) and ray(b,P) such that the following three conditions hold: (i) ray(a,P) and ray(b,P) intersect at some point x, (ii) x is a vertex of P, and (iii) P is contained in Δabx.

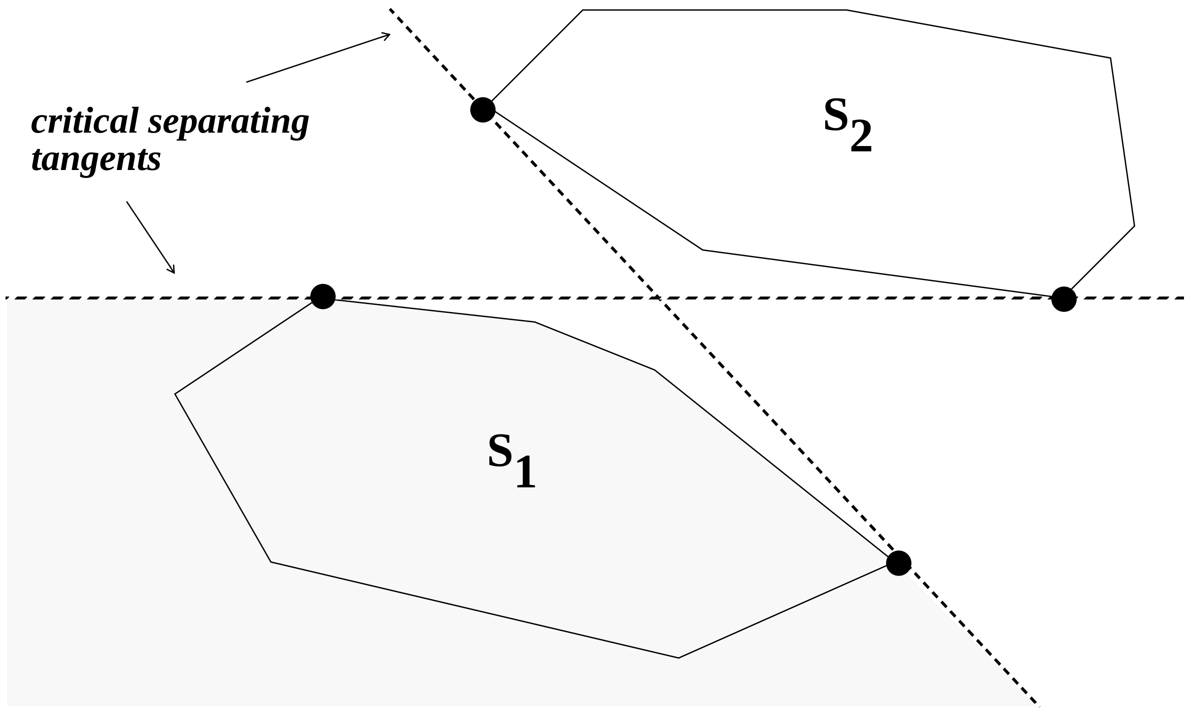

Fig. 4.5 Illustrating the *maximal linearly separable extension* (shaded region) of one convex polygon S_1 with respect to another S_2.

time. The algorithm follows from the results presented in section 2 and a few straightforward lemmas that we mention here without proofs.

For any integer $n \geq 3$, we define a *polygon* in the Euclidean plane E^2 as the figure $P = [x_1,x_2,...,x_n]$ formed by n points $x_1,x_2,...,x_n$ in E^2 and n line segments $[x_i,x_{i+1}]$, i=1,2,...,n-1, and $[x_n,x_1]$. The points x_i are called the *vertices* of the *polygon* and the line segments are termed its *edges*. We assume the vertices of P are in *general position*, i.e., no three vertices are colinear and that the polygon is in *standard* form, i.e., the vertices appear in counterclockwise order as their index increases. A polygonal sub-chain of P will be denoted by $C[x_i,x_{i+1},...,x_{j-1},x_j]$. Therefore $C[x_1,x_2,...,x_n]$ corresponds to P with the segment $[x_n,x_1]$ removed. A polygon P is called a *simple* polygon provided that no point of the plane belongs to more than two edges of P and the only points of the plane that belong to precisely two edges are the vertices of P.

A simple polygon has a well defined interior (denoted by *int*(P)) and exterior (denoted by *ext*(P)). We will follow the convention of including the interior of a polygon when referring to P.

Let P be a convex polygon. Denote the closed half-plane to the left of a directed line determined by two ordered points x,y by *HL*(x,y). The corresponding closed half-plane to the right of a directed line determined by two ordered points x,y is denoted by *HR*(x,y). It is also convenient to denote the edge $[p_i,p_{i+1}]$ of P by e_i. Thus $HR(e_i)$ is the outer half plane determined by edge e_i.

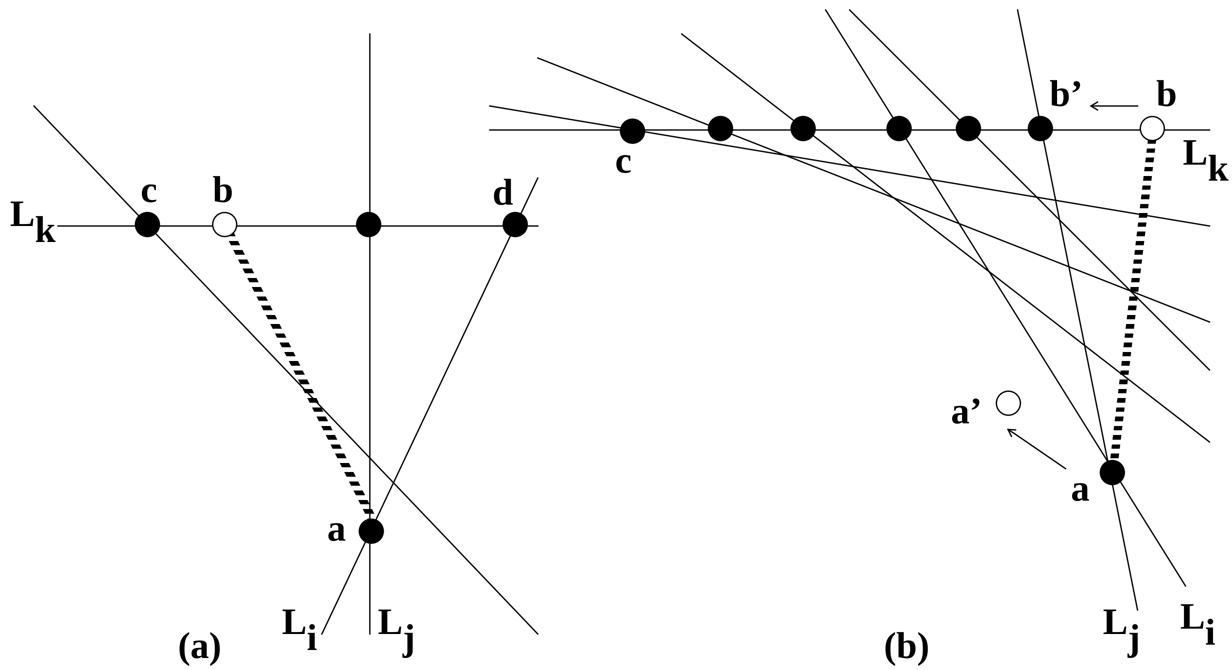

Fig. 4.4 Illustrating case 2 in the proof of Lemma 4.1.

each edge in turn the latter problem can be solved in O(n^2) time. Sack and Suri [SS86] discovered a linear-time algorithm for determining all (if any) such edges of a given polygon. Recently Yan Ke [Ke88] considered the problem of detecting the weak visibility of a simple polygon from an internal line segment. He presents an O(n log n) time algorithm that tests if a polygon is weakly visible from some internal line segment and reports such a line segment if it exists. He also addresses the query version of this problem: given a query line segment in P, is P weakly visible from it? He shows that this question can be answered in O(log n) time after the polygon is preprocessed in O(n log n) time using O(n) space. More relevant to the work in this paper, Ke shows that the shortest such line segment can be found in O(n log n) time. All these results are concerned with *internal* visibility. Bhattacharya, Kirkpatrick and Toussaint [BKT89] have considered the corresponding computational problems for the case of *external weak visibility*. A polygon P is said to be *weakly-externally-visible* provided that for every point *x* on the boundary of P there exists an infinite ray starting at *x* that intersects P only at *x*. More intuitively speaking P is weakly-externally-visible if when a guard patrols along a circle containing P in its interior then the entire boundary of P is visible by the guard at one time or another. In particular they show that given a simple polygon P, all lines not intersecting P from which P is weakly externally visible can be found in O(n) time. Furthermore queries can be answered in O(log n) time after the polygon is preprocessed in O(n) time using O(n) space. We note that external visibility from finite sets of points rather than lines or line segments has also received attention in the mathematics literature [BV76], [Va70] where the results are of a combinatorial nature.

In this section we consider the question of computing the shortest line segment from which a given convex polygon is weakly *externally* visible. It is shown that, given a *convex* polygon P, the *minimal length* line segment from which P is *weakly externally visible* can be found in O(n)

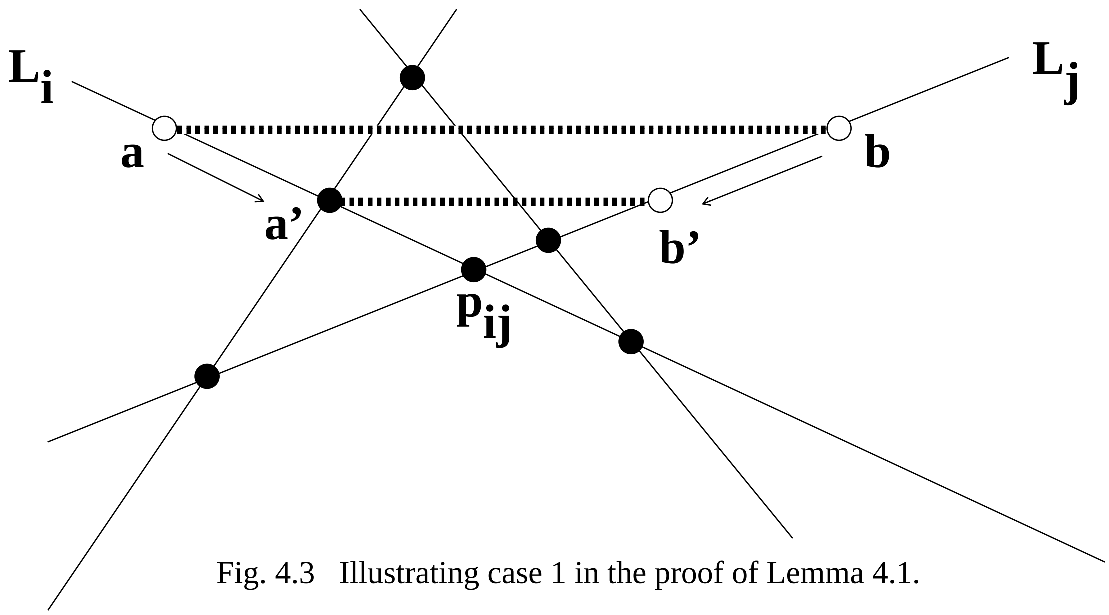

Fig. 4.3 Illustrating case 1 in the proof of Lemma 4.1.

rectangle induces a halfplane, namely, that which contains the rectangle in question and has its bounding line colinear with the said edge. Another such halfplane is induced by the highest lower edge of some rectangle. Consider the boundary of the intersection of these two halfplanes. Now intersect this boundary with the complement of $int(E_{max}(\boldsymbol{F}_t/\boldsymbol{F}_b)) \cup int(E_{max}(\boldsymbol{F}_t/\boldsymbol{F}_b))$. It is straightforward to compute this structure in linear time. Furthermore, the envelopes consist of at most two edges each reducing the problem to eight instances of the geometric optimization problem considered in section 2. Let k denote the maximal cardinality of the convex hulls of each of the four sets of corners of the rectangles in $\boldsymbol{F}$. In other words, $4k$ is an upper bound on the combinatorial complexity of the space of transversals for $\boldsymbol{F}$. Then we have the following theorem.

Theorem 4.4: Given a set $\boldsymbol{F} = \{F_1, F_2, ..., F_n\}$ of n isothetic rectangles in $\boldsymbol{R}^2$ the minimal length line segment that intersects every member of $\boldsymbol{F}$ can be computed in O(n log k) time and O(n) space.

5. Applications

5.1. Minimal Sets of External Visibility

The notion of weak visibility has also received attention in both the mathematics and computer science literatures. Horn and Valentine [HV49] have characterized L-sets in terms of their weak visibility properties while such characterizations for convex and star-shaped sets have been obtained by Shermer and Toussaint [ST88]. Avis and Toussaint [AT81] showed that given a simple polygon P and a specified edge **e** of P, whether P is edge-visible from **e** can be determined in O(n) time. A polygon P is *edge-visible* from an edge **e** if for every point x in P there exists a point y in **e** such that the line segment xy is in P. A more difficult problem is to determine whether there exists an edge of P from which P is edge-visible. Clearly by applying the algorithm in [AT81] to designed to test whether any two segments intersect. Therefore we have the following result.

Corollary 4.2: Given a set $S = \{s_1, s_2,...,s_n\}$ consisting of n line segments, the minimal length line segment that intersects S can be computed in O(n log n) time and O(n) space if the line segments do not intersect.

4.4. The Case of Isothetic Rectangles

Transversals are closely related to the notion of separability [We88] through which we take a small detour. Let S_1 and S_2 be two finite sets of points in $\boldsymbol{R}^2$. We say that S_1 and S_2 are *linearly separable* if there exists a line that partitions the plane into two closed half-planes H_1 and H_2 such that $S_1 \subset H_1$ and $S_2 \subset H_2$. A point $x \in \boldsymbol{R}^2$ is a *linearly separable extending* (LSE) point of S_1 with respect to S_2 if, (1) $x \notin S_1$, and (2) any line that separates S_1 from S_2 also separates x from S_2. The *maximal linearly separable extension* of S_1 with respect to S_2, denoted by $E_{max}(S_1/S_2)$, is the union of S_1 with all points of $\boldsymbol{R}^2$ that are *linearly separable extending* points of S_1 with respect to S_2. The boundary of $E_{max}(S_1/S_2)$ is an unbounded convex polygonal chain. Let S* denote the complement of $int(E_{max}(S_1/S_2)) \cup int(E_{max}(S_2/S_1))$. Clearly, S* is the union of all separating lines of S_1 and S_2 and its boundary is a description of all such lines (see Fig. 4.5). Thus the cardinality of S* is the combinatorial complexity of the *space* of an equivalence class of transversals.

Given a set $\boldsymbol{F} = \{F_1,F_2,...,F_n\}$ of n isothetic rectangles in $\boldsymbol{R}^2$, it is desired to find the minimal-length line segment that intersects every member of $\boldsymbol{F}$ if it exists. The problem of whether such an $\boldsymbol{F}$ admits a *common line transversal* was originally investigated by Edelsbrunner [Ed85]. Let $\boldsymbol{F}_t = \{t_1,t_2,...,t_n\}$ denote the set of north-east vertices of the members of $\boldsymbol{F}$. Similarly let $\boldsymbol{F}_b = \{b_1,b_2,...,b_n\}$ denote the set of south-west vertices of the members of $\boldsymbol{F}$. Finally let $\boldsymbol{F}_l$ and $\boldsymbol{F}_r$ denote, respectively, the set of north-west and south-east vertices of the members of $\boldsymbol{F}$. See Fig. 4.6 for an illustration of these sets. Edelsbrunner showed that whether $\boldsymbol{F}$ admits a common transversal can be determined in O(n) time by reducing the problem to a separability question. The key observation is that $\boldsymbol{F}$ admits a common transversal if, and only if, either $\boldsymbol{F}_t$ and $\boldsymbol{F}_b$, or $\boldsymbol{F}_l$ and $\boldsymbol{F}_r$, are linearly separable. Furthermore, the *set-set-separation* problem can be solved with linear programming. Thus it suffices to apply one of the linear-time linear-programming algorithms of either Megiddo [Me83] or Dyer [Dy84] to determine whether a separator exists. If no separator exists for either of the two problems then we conclude $\boldsymbol{F}$ does not admit a line transversal. If at least one of the two problems admit a separator then we compute the respective convex hulls and critical lines of support as in the previous sections. Here it is advantageous to compute the convex hull with the output-size sensitive algorithm of Kirkpatrick & Seidel [KS86] in O(n log *h*) time where *h* is the number of vertices on the convex hull. Each problem induces only a single equivalence class of transversals and furthermore it is easy to construct its left and right envelopes in linear time. Consider the left envelope for the sets $\boldsymbol{F}_t$ and $\boldsymbol{F}_b$ in Fig. 4.6. The leftmost right edge of a

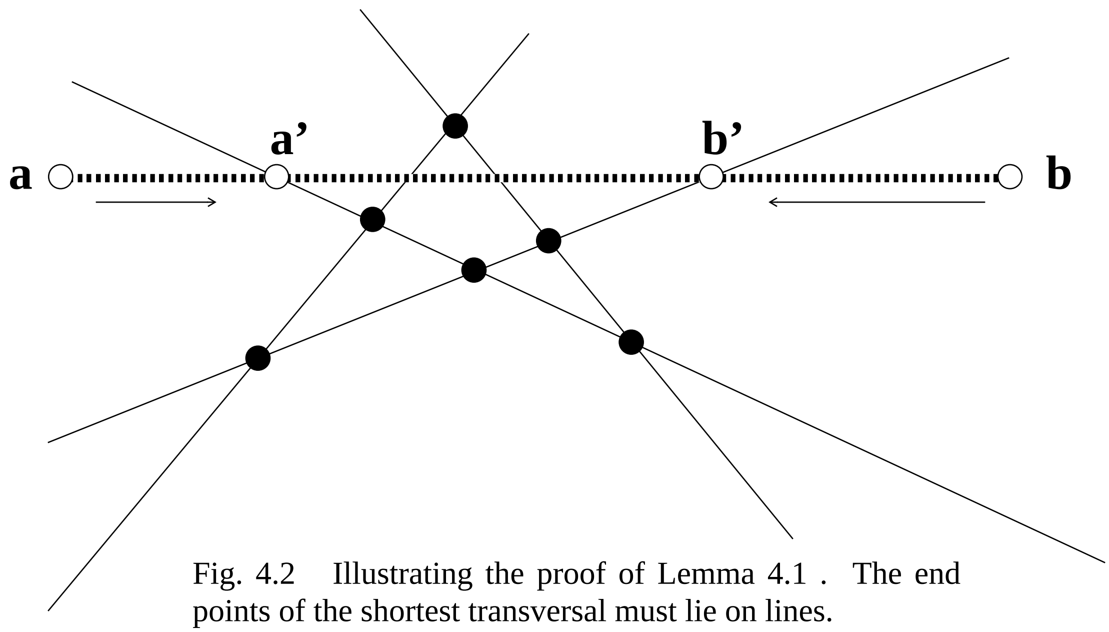

Fig. 4.2 Illustrating the proof of Lemma 4.1 . The end points of the shortest transversal must lie on lines.

4.3. The Case of Non-intersecting Line Segments

It should be clear from the previous discussion that when S = $\{s_1, s_2,...,s_n\}$ consists of a set of non-intersecting line segments the left and right envelopes for each equivalence class of transversals is made up of a portion of a single line segment. Therefore all the machinery used in the previous section for efficiently updating the envelopes is superfluous. Furthermore, under these restrictive assumptions we do not require the nested binary search algorithm for computing the constrained visible distance problem. Since these are the only portions of the algorithm requiring O(n $\log^2$ n) time, we have the following theorem.

Theorem 4.3: Given a set S = $\{s_1, s_2,...,s_n\}$ consisting of n non-intersecting line segments the minimal length line segment that intersects S can be computed in O(n log n) time and O(n) space.

O'Rourke [O'R81] showed that given a set of data ranges (vertical line segments in sorted order as a function of time) line-fitting between these data ranges can be done in linear time. Since data ranges are a special case of line segments which are non-intersecting, our results imply the following.

Corollary 4.1: Given a set S=$\{s_1, s_2,...,s_n\}$ consisting of n data-ranges in sorted order, the minimal length line-segment fit through S can be computed in O(n) time and space.

Chazelle & Edelsbrunner [CE88] recently showed that given n line segments in the plane all *k* pairwise intersections can be computed in O(n log n + *k*) time and O(n + *k*) space. By running this algorithm and stopping as soon as the first intersection is found we can detect if S consists of pairwise non-intersecting line segments in O(n log n) time and O(n) space. Alternately, we can use the much simpler O(n log n) time line-sweep algorithm of Shamos and Hoey [SH76] specifically

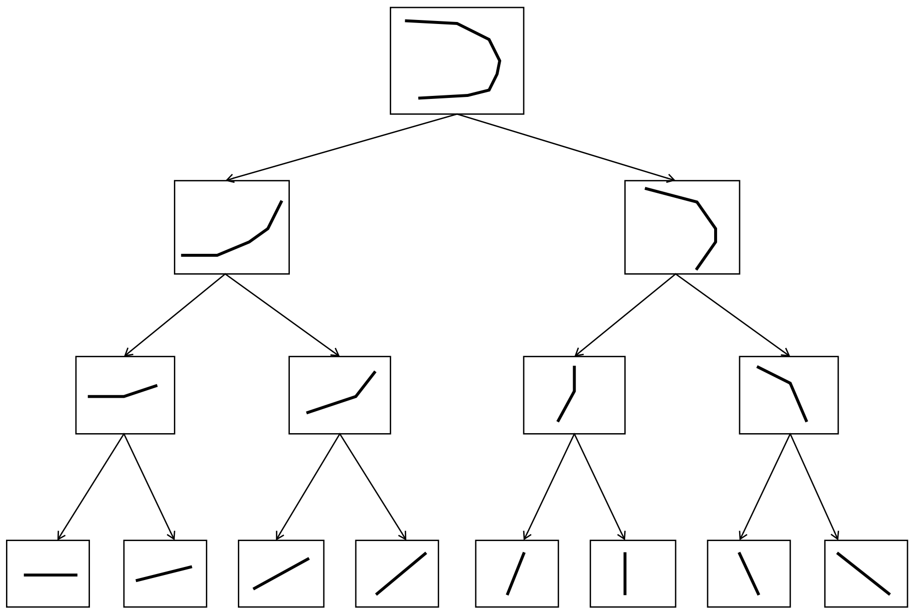

Fig. 4.1 Illustrating the data-structure used in the efficient updating of left and right envelopes corresponding to the transitions between equivalence classes of transversals.

CH(**I**) and therefore **b** ∈ CH(**I**) which contradicts the assumption of Case 2.

Case 2.2.2: All points of **I** lie on one side (say to the left of the directed line through **a**,**b**) of **b** on line L_k. Let **b'** be the point of **I** on L_k closest to **b** and let **c** be the point of **I** on L_k furthest from **b** and refer to Fig. 4.4 (b). Consider triangle [**b**, **a**, **c**]. Since all lines in **L** other than L_k intersect L_k to the left of **b**, it follows that none of these lines intersect the open line segment (**a**, **c**). For otherwise [**a**, **b**] would not be a transversal, contrary to the assumption of the lemma. Let **a'** ∈ (**a**, **c**) denote the point such that [**a'**, **b'**] is parallel to [**a**, **b**]. By the method of construction of [**a'**, **b'**] it follows that it is a transversal of **L** and it is shorter than [**a**, **b**], a contradiction. Q.E.D.

This lemma allows us to transform the problem of computing the shortest transversal of a set of lines to the problem of finding the shortest transversal of a set of line segments in O(n log n) time. First we compute the convex hull of **I** in O(n log n) time with the algorithm of either Ching & Lee [CL85] or Atallah [At86]. Then we intersect each line in **L** with the resulting convex polygon in logarithmic time per line using the algorithm of Chazelle & Dobkin [CD80]. Although **I** contains $O(n^2)$ points the CH(**I**) contains only O(n) points in the worst case [CL85], [At86] and O(1) points on the average for almost any definition of a random arrangement of lines and for almost any distributions on the resulting parameters [DT90]. Therefore the set of line segments can be computed in O(n log n) time. We thus have the following theorem.

Theorem 4.2: Given a set of lines **L** = $\{L_1, L_2, ..., L_n\}$ the shortest line segment that intersects every member of **L** can be found in $O(n \log^2 n)$ time and O(n) space.

4.2. The Shortest Transversal of an Arrangement of Lines

Let **L** = {L_1, L_2,...,L_n} denote the set of lines where each line L_i is specified by an equation $Y=a_iX + b_i$ for some real numbers a_i, b_i, i=1,2,...,n. Let p_{ij} denote the intersection point of L_i and L_j. The set **I** = { p_{ij} | $1 \leq i,j \leq n$ } denotes the set of intersection points. Finally, let CH(**I**) denote the convex hull of **I**.

Lemma 4.1: The shortest transversal of a set of lines **L** is identical to the shortest transversal of the set of line-segments which are obtained by intersecting each given line in **L** with the convex hull of the intersection points determined by these lines.

Proof: The shortest transversal of **L** clearly has the property that its end points each belong to some line of **L**. For assume the contrary and let **a** and **b** denote its end points as illustrated in Fig. 4.2. Let **a'** denote the first point of intersection of a point travelling from **a** to **b** along segment [**a**,**b**] that it makes with a line in **L**. Similarly, let **b'** denote the first point of intersection of a point travelling from **b** to **a** along segment [**a**,**b**] that it makes with a line in **L**. The new line segment [**a'**,**b'**] is a transversal because it intersects all the lines that [**a**,**b**] does and it is shorter than [**a**,**b**] which is a contradiction. The same argument shows that the shortest transversal of a set of line segments must have each of its end points on some line segment. Furthermore, by the method of construction of the line segments from **L**, these must all lie in CH(**I**). Hence the statement of the lemma implies that the end points of the shortest transversal for **L** must lie in CH(**I**). Conversely, if the shortest transversal for **L** lies in CH(**I**) then we need not examine the portions of lines in **L** lying outside CH(**I**). Therefore let [**a**,**b**] be the shortest transversal of **L** and assume that **a** and **b** do not both lie in CH(**I**). We will look for a contradiction. Two cases arise.

Case 1: Both **a** and **b** $\notin$ CH(**I**). Neither **a** nor **b** may lie on points of **I** or they would lie in CH(**I**). Therefore, without loss of generality let **a** lie in line L_i and let **b** lie on line L_j. These two lines intersect at p_{ij} and this point together with **a** and **b** determine a triangle illustrated in Fig. 4.3. Let **a** and **b** each travel towards p_{ij} on their respective lines L_i and L_j such that the resulting line segment remains parallel to [**a**,**b**] until the line segment intersects an intersection point of **I** on either L_i or L_j. Without loss of generality assume **a'** $\in L_i$ is such a point and let **b'** $\in L_j$ be the other endpoint of the resulting line segment. By the method of construction of [**a'**,**b'**] it follows that it must be a transversal and its length is shorter than that of [**a**,**b**], which is a contradiction.

Case 2: **a** $\in$ CH(**I**) and **b** $\notin$ CH(**I**). Therefore **b** cannot lie on an intersection point of **I**. We thus have two sub-cases.

Case 2.1: Point **a** $\notin$ **I**. In this case both **a** and **b** each lie on distinct and unique lines in **L** and a similar argument to that used in case 1 leads to a contradiction.

Case 2.2: Point **a** $\in$ **I**. Let **a** lie in lines L_i and L_j and let **b** lie on line L_k and refer to Fig. 4.4. Two sub-cases arize depending on whether or not points of **I** lie on both sides of **b** on line L_k.

Case 2.2.1: Points of **I** lie on both sides of **b** on line L_k. Let **c** $\in$ **I** lie on one side of **b** and **d** $\in$ **I** lie on the other side of **b** on line L_k and refer to Fig. 4.4 (a). By convexity triangle [**a**, **c**, **d**] $\in$

describe the algorithm for the case of left envelopes only.

We describe first an O(n log n) time algorithm for computing the left envelope of S for region R_1. The data structure created in this stage will allow the computation of the left envelope for R_2 by affording insertions and deletions which take $O(\log^2 n)$ time per update.

Let L_w be a line transversal that is a member of the equivalence class corresponding to R_1. Let θ_i be the angle of the line segment s_i directed from $b(s_i,L_w)$ to $a(s_i,L_w)$ with the positive x-direction. First we sort all the line segments in S according to increasing values of θ. Then we apply a divide-and-conquer strategy to this sorted list. The merging step involves intersecting two counterclockwise directed convex chains that have the special property that the angles of their directed edges with respect to the positive x-direction of one chain are all greater than (or all less than) that of the other chain. As a result these two chains, if they intersect, do so at most once. Whether these two chains intersect can be determined in logarithmic time by applying the hierarchical techniques developed in [DK83]. If the chains do not intersect then we need only keep the *leftmost* of the two chains. We can determine which chain is the leftmost chain by intersecting both chains with the *witness* transversal L_w in logarithmic time using the algorithm of Chazelle & Dobkin [CD80]. If the chains do intersect we can find the intersection point in logarithmic time by again applying the techniques in [DK83]. Hence O(n log n) time is sufficient to compute the left envelope of R_1.

This algorithm creates a balanced tree [AHU83] with the leaves containing the directed line segments in sorted order by angle as illustrated in Fig. 4.1. An internal node represents the left envelope of the line segments in its subtree. Note that if we were to store the left envelope explicitly in each node the total storage space requirement for this tree would be O(n log n). However, space requirements can be reduced to O(n) by storing at each internal node only that part of the chain which is not present in the left envelope of its parent node with methods such as those discussed in [OV81]. The root node contains the left envelope of S for region R_1.

We now see what happens when the end points of a line segment s_i switch to create a new equivalence class. First we delete the directed line segment s_i with angle θ_i from the tree. This operation will require O(log n) time to update the information stored at each node along the path from the root node to the leaf containing s_i by merging at each node the chain with the missing edge with its brother to create a new parent. Since the depth of this tree is O(log n) this step must be done O(log n) times. Thus the deletion step requires $O(\log^2 n)$ time. We then insert a directed line segment with angle $\theta_i+180°$ (modulo 360°). This step also requires updating left envelopes of O(log n) nodes leading to a total time of $O(\log^2 n)$. According to Lemma 3.2 there are at most n switches. We have therefore established the following theorem.

Theorem 4.1: Algorithm *SHORTEST-TRANSVERSAL* finds the minimal length line segment that intersects a given set of n line segments in $O(n \log^2 n)$ time and O(n) space.

Algorithm *SHORTEST-TRANSVERSAL*

Input: A set $S = \{s_1, s_2,...,s_n\}$ of line segments in $\boldsymbol{R}^2$ given as pairs of end points with their associated cartesian coordinates.

Output: The line segment of minimum length that intersects every member of S or the conclusion that S does not admit a transversal of any length.

Begin:

Step 1: Transform each line segment of S into a double wedge in the dual plane.

Step 2: Compute the intersection **R**, of $R_1, R_2,..., R_k$ of the double wedges.

Step 3: If the intersection is empty ***exit*** with *"there exists no transversal."*

Step 4: For each convex region $R_i \subseteq \mathbf{R}$, i=1,2,...,k do the following:

(a) Compute the left envelope, the right envelope, the convex hull of the upper end points of S, and the convex hull of the lower end points of S.

(b) Compute the critical lines of support between the two convex hulls and concatenate their relevant portions with the envelopes and convex hull boundaries to form a *kite-shaped* polygon.

(c) Determine the minimum visible distance between the left and right chains of the kite-shaped polygon.

Step 5: Select the smallest minimum distance encountered in Step 4 and ***exit*** with the line segment that determines this distance as the shortest transversal of S.

end

Steps 1-3 of Algorithm *SHORTEST-TRANSVERSAL* can be computed in O(n log n) time with the algorithm of Edelsbrunner et al. [EMPRWW82]. In order to implement Step 4 efficiently we must be able to avoid recomputing the left envelope, the right envelope, the convex hull of the upper end points of S, and the convex hull of the lower end points of S from scratch for every region $R_i \subseteq \mathbf{R}$, i=1,2,...,k. Actually, for each i=1,2,...,k the lower hull of the top convex hull and the upper hull of the bottom convex hull can be obtained from the boundary information of R_i in the dual plane in time proportional to the cardinality of the boundary of R_i. Therefore all the upper and lower relevant subchains of the top and bottom convex hulls can be obtained in O(n) time once **R** is computed in Step 2. The critical support lines between the pair of convex hulls corresponding to each region R_i can be found with the "rotating-calipers" [To83] also in time proportional to the cardinality of the boundary of R_i or even in logarithmic time if desired [Ro85]. Therefore all the lower and upper chains of the kite-shaped polygons corresponding to the R_i can be computed in linear time after Step 2 is executed. Thus we concentrate on computing efficiently the left and right envelopes correponding to R_i from the left and right envelopes of R_{i-1}. Without loss of generality we

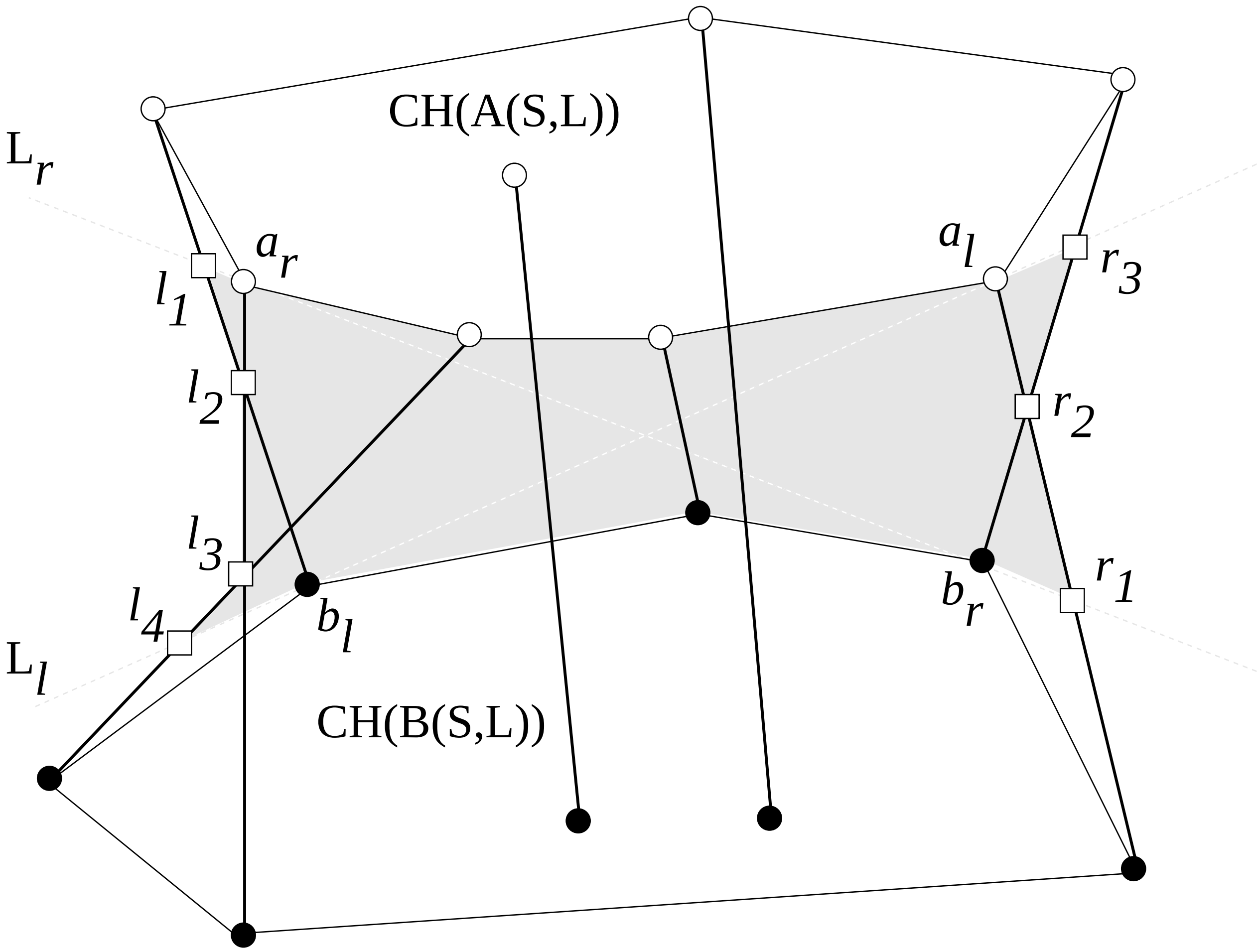

Fig. 3.4 Illustrating the left envelope (l_1,l_2,l_3,l_4), the right envelope (r_1,r_2,r_3), the top convex hull CH(A(S,L)), the bottom convex hull CH(B(S,L)), and the relevant kite-shaped polygon induced by these four polygonal chains (shaded).

4. Algorithms for Computing Shortest Transversals

4.1. The General Case

First we give a high-level outline of the algorithm for computing the shortest transversal of a given set of line segments.

Lemma 3.3: The end points of the shortest transversal s for a given equivalence class of transversals must lie on the left and right envelopes of that equivalence class and s cannot intersect the interior of CH(A(S,L)) and CH(B(S,L)).

Let L_r denote the critical line of support separating CH(A(S,L)) and CH(B(S,L)) with extremal clockwise rotation and let a_r and b_r denote the tangent vertices of CH(A(S,L)) and CH(B(S,L)), respectively, that are furthest apart (see Fig. 3.4). Similarly, let L_l denote the critical line of support separating CH(A(S,L)) and CH(B(S,L)) with extremal counter-clockwise rotation and let a_l and b_l denote the tangent vertices of CH(A(S,L)) and CH(B(S,L)), respectively, that are furthest apart. Let *bd* denote the boundary of a set and let *bd(*CH(A(S,L))*)* be specified by its vertices listed in counter-clockwise order. Then in lemma 3.3 we need only concern ourselves with those portions of CH(A(S,L)) and CH(B(S,L)) determined by their critical support lines. More specifically, we are only concerned with the convex polygonal chains $[a_r,...,a_l]$ from *bd(*CH(A(S,L))*)* and $[b_r,...,b_l]$ from *bd(*CH(B(S,L))*)*. In fact, the region of interest is a *kite-shaped* polygon obtained by concatenating the left and right envelopes and the chains $[a_r,...,a_l]$ and $[b_r,...,b_l]$ with the portions of the critical support lines that connect the upper and lower chains with the envelopes. For example, in Fig. 3.4, the relevant kite-shaped polygon (shaded) is given by $[r_1, r_2, r_3] \cup [r_3, a_l] \cup [a_l ,..., a_r] \cup [a_r, l_1] \cup [l_1, l_2, l_3, l_4] \cup [l_4, b_l] \cup [b_l ,..., b_r] \cup [b_r, r_1]$, where the four convex vertices in question are $r_1, r_3, l_1,$ *and* l_4. We have thus transformed the problem of computing the shortest transversal for a given equivalence class of transversals into the problem of computing the minimum visible distance between two opposite concave chains of a kite-shaped polygon that is easily obtained from the available information.

Lemma 3.4: Given a simple polygon of n vertices known to be kite-shaped, with its four convex vertices available in order, the minimum visible distance between a pair of opposite concave chains can be computed in $O(\log^2 n)$ time.

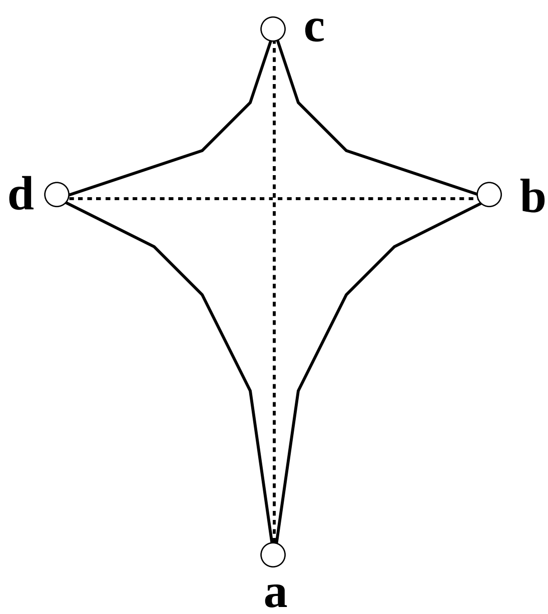

Fig. 3.3 Illustrating a *kite-shaped* polygon.

The proof is too lengthy to reproduce here and forms the topic of a companion paper [BET91]. Here we provide only the basic idea. The approach is to first disregard the upper and lower chains in computing the minimum distance between the left and right chains. If this unconstrained solution yields a line segment (representing the distance) that does not intersect the upper or lower chains then it is the final solution. If it does intersect the upper (lower) chain then the final solution is tangent to the upper (lower) chain and can be found by an analogous but more sophisticated prune-and-search method.

computing. Let L_s be the transversal which contains s. Clearly L and L_s are members of the same equivalence class. In what follows it will be useful to define a new class of polygons as follows.

Definition: A simple polygon P is a *kite* (kite-shaped) provided that it contains only four convex vertices labelled a,b,c,d in order and such that the diagonals [a,c] and [b,d] lie in the polygon (see Fig. 3.3).

Definition: A line segment s_i is *left* (*/right*) of another line segment s_j , with respect to a transversal L, if the intersection of s_i with L lies to the *left* (*/right*) of the intersection of s_j with L.

Definition: The *left-envelope* of S with respect to an equivalence class of transversals is defined as the union of all points p such that p is the leftmost intersection point of a line segment in S with some transversal belonging to the equivalence class.

The *right-envelope* of S is defined in a similar manner. Figure 3.4 illustrates the left and right envelopes for an equivalence class of transversals for a set of line segments. It is straightforward to verify that these envelopes are convex polygonal chains composed of a concatenation of portions of the line segments of S. Let CH(A(S,L)) and CH(A(S,L)) denote, respectively, the convex hulls of the sets A(S,L) and B(S,L). We shall refer to these as the *top* and *bottom* convex hull, respectively. Then the following lemma is easily established and stated without proof.

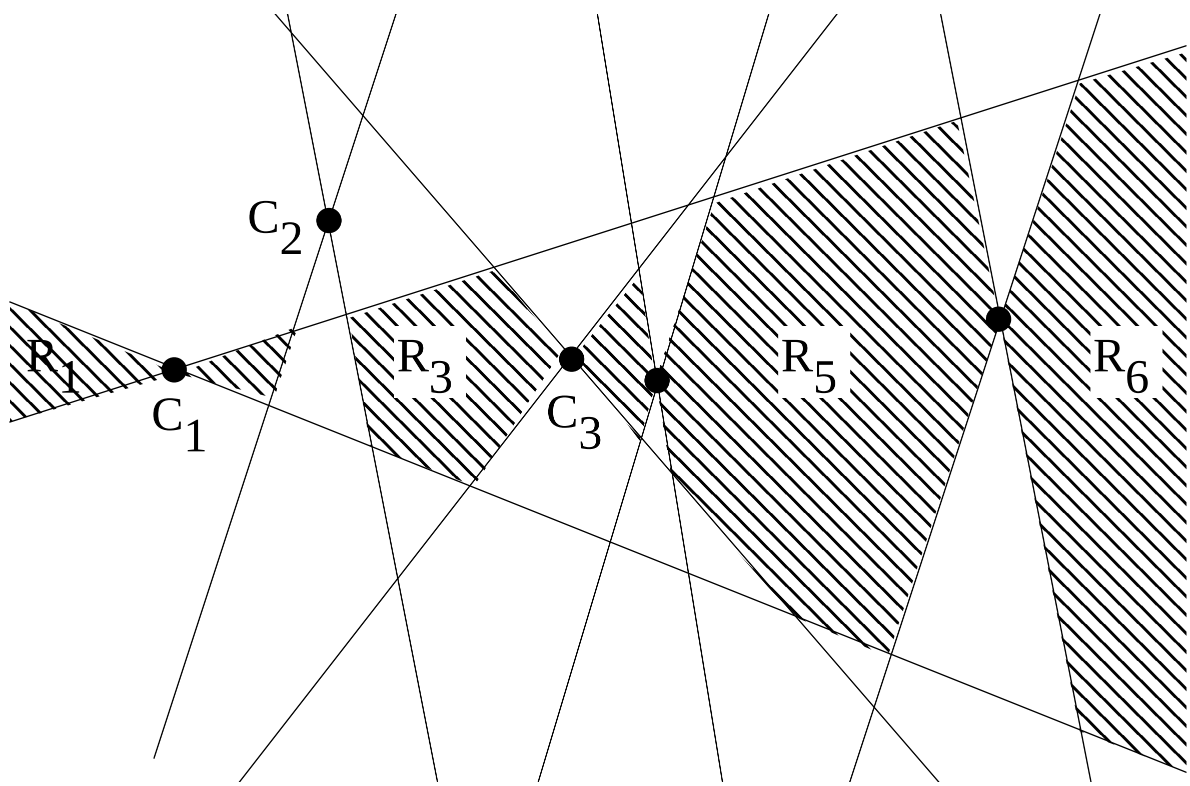

Fig. 3.2 Illustrating the convex regions (in the dual plane) of the intersection of five double wedges induced by five line segments in the primal plane. The convex regions are shaded and the solid dots mark the *centers* of the double wedges.

Proof: It has been shown in [EMPRWW82] that the intersection of the n double wedges consists of at most n+1 convex polygons monotonic in the x-direction. Every point in this intersection corresponds to a transversal for S. From property (f) above it follows that there is a one-to-one correspondence between the convex polygons and the equivalence classes of transversals. Q.E.D.

Let R_i and R_{i+1} be two adjacent convex regions in the intersection of the double wedges. Then there are only two structural possibilities [EMPRWW82] for their location: (a) R_i and R_{i+1} share a vertex which is a center of some double wedge or (b) R_i and R_{i+1} are disjoint. Let $L(R_i)$ and $L(R_{i+1})$ be two representative transversals corresponding to points in R_i and R_{i+1}, respectively. When R_i and R_{i+1} share a center of a double wedge of some line segment, say s_k, then $A(S,L(R_{i+1})) = \{A(S,L(R_i)) - a(s_k,L(R_i))\} \cup \{b(s_k,L(R_i))\}$ and $B(S,L(R_{i+1})) = \{B(S,L(R_i)) - b(s_k,L(R_i))\} \cup \{a(s_k,L(R_i))\}$. This suggests that we may be able to compute $A(S,L(R_{i+1}))$ and $B(S,L(R_{i+1}))$ from $A(S,L(R_i))$ and $B(S,L(R_i))$, respectively, in an efficient manner. In addition, when R_i and R_{i+1} are disjoint, the line segments whose end points switch sides from above (/below) of $L(R_i)$ to below (/above) of $L(R_{i+1})$ are precisely those whose corresponding double wedge *centers* lie between regions R_i and R_{i+1}. Furthermore, once the end points of a line segment have switched sides they will never switch sides again. We therefore have the following straightforward lemma which we state without proof.

Lemma 3.2: As one traverses the arrangement determined by the n double wedges in the positive x-direction the number of end-point switches that occurs is at most n.

It is a simple matter to determine in linear time the shortest line segment that intersects a set of line segments S if we demand that it be parallel with a given transversal L of S. We thus concentrate now on the next most difficult problem: that of computing the shortest transversal from amongst a single equivalence class of transversals. Accordingly, let L be a transversal belonging to such a class and let s be the shortest line segment of the equivalence class we are interested in

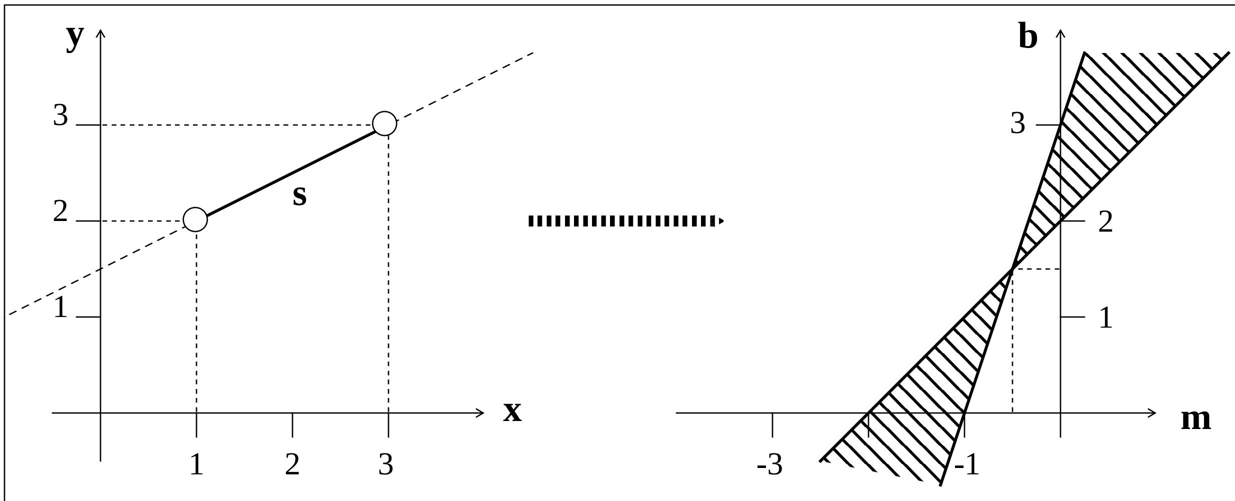

Fig. 3.1 A line segment s in the *primal* plane and its transformed *double-wedge* (shaded) in the *dual* plane.

through point p such that either the perimeter or the area of triangle OHK is minimized are in fact classical fundamental problems that have simpler and more elegant solutions [La59] than our problem considered here and have been applied successfully to several interesting problems in computational geometry [CY84], [DA84], [KL85], [OAMB86]. Finally, we remark that Lemma 2.2 holds for the more general case when OQ and OR are themselves diverging convex chains. This more general version of the lemma will be used in subsequent sections.

3. Geometric Preliminaries for Computing Shortest Transversals

Let $S = \{s_1, s_2, ..., s_n\}$ denote the given set of line segments in $\boldsymbol{R}^2$ (termed the *primal* plane). We first describe briefly the technique of Edelsbrunner et al. [EMPRWW82] to construct a description of all infinite length transversals admitted by S. Each endpoint (x,y) of a line segment in S in the *primal* plane is transformed to the line $b=xm+y$ in the *dual* plane and each line $y=kx+d$ is transformed into a point $(-k,d)$. Thus each line segment in the *primal* plane is transformed to a double wedge in the *dual* plane as illustrated in Fig. 3.1. Furthermore, a point in the double wedge in the dual plane represents a transversal of the corresponding line segment in the primal plane. In this way the problem of determining a description of all transversals in the primal plane is converted to the problem of intersecting a set of double wedges in the dual plane. The following properties are shown in [EMPRWW82] and illustrated in Fig. 3.2.

(a) the intersection of the double wedges consists of at most n+1 convex regions,

(b) the cardinality of the intersection is at most 8n+8,

(c) the intersection region is monotonic in the x-direction,

(d) a point in the intersection region corresponds to a transversal for S,

(e) vertical lines through the centers of the double wedges do not intersect the interior of the intersection region.

(f) the end points, of line segments of S, which lie above a transversal L in the primal plane are precisely those whose corresponding dual lines intersect the vertical half-lines emanating from the dual point of L in the positive direction.

Let L be a transversal of S and assume it is not vertical. Let $a(s_i,L)$ and $b(s_i,L)$ denote the end points of line segment s_i that lie *above* and *below*, respectively, of the transversal L. Let $A(S,L) = \{a(s_1,L), a(s_2,L),..., a(s_n,L)\}$ denote the set of end points of S that lie above L. Similarly let $B(S,L) = \{b(s_1,L), b(s_2,L),..., b(s_n,L)\}$ denote the set of end points of S that lie below L.

Definition: Two transversals L_1 and L_2 are called *equivalent* if $A(S,L_1) = A(S,L_2)$. All transversals that are pairwise equivalent form an *equivalence class* of transversals.

Lemma 3.1: A set of line segments $S = \{s_1, s_2,...,s_n\}$ admits at most n+1 distinct equivalence classes of transversals.

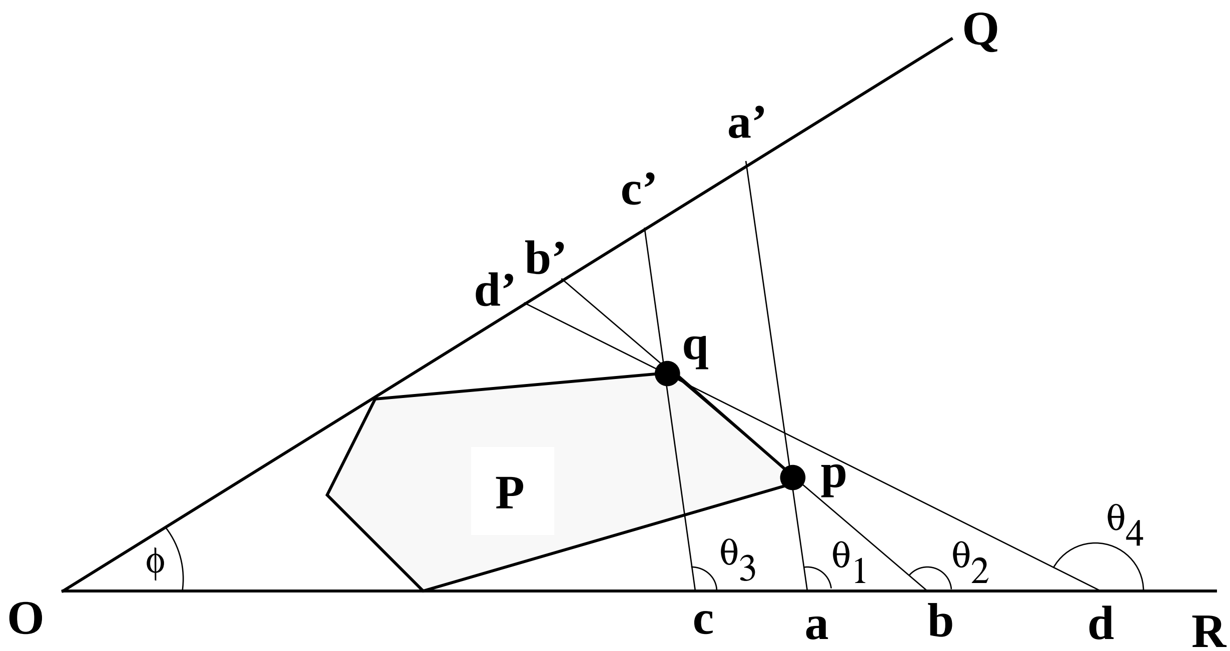

Fig. 2.5: Illustrating the proof of lemma 2.2: case 2: *l*(aa’) is decreasing at $\theta = \theta_1$.

Subcase 2.2.1: p $\notin$ *int*(ΔOHK)

Let HK* be cc’ making angle $\theta = \theta_3$ with OR. By the unimodality of *l*(HK(q,θ)) from lemma 2.1 it follows that *l*(cc’) < *l*(bb’) and *l*(HK(q,θ)) increases continually as θ varies from θ_3 past θ_2 and on to $\theta = \pi$. This implies that *l*(HK(P,θ)) reaches its minimum value at $\theta = \theta_2$ when HK = bb’ and that HK(q,θ) for $\theta = \theta_2 + \varepsilon$, $\varepsilon > 0$, is an instance of *Case 1*. Thus in this subcase *l*(HK(P,θ)) continually increases for $\theta_2 < \theta < \pi$ and *l*(bb’) is the minimum of *l*(HK(P,θ)) in the interval $\phi < \theta < \pi$.

Subcase 2.2.2: p $\in$ *int*(ΔOHK)

Let HK* be dd’ making angle $\theta = \theta_4$ with OR. By the unimodality of *l*(HK(q,θ)) from lemma 2.1 it follows that *l*(dd’) < *l*(bb’) and *l*(HK(q,θ)) decreases continually as θ varies from θ_2 to θ_4. Therefore *l*(HK(P,θ)) continues to decrease as θ varies from θ_1 about p to θ_4 about q.

Thus we have shown that for a starting vertex p of P and corresponding angle θ_1, *l*(HK(P,θ)) either continuously increases, or first decreases and then increases as θ varies from θ_1 to π. Since this holds for any p and any corresponding angle θ_1 the statement of the lemma follows. Q.E.D.

We remark in closing this section that the related problems of finding the line segment HK

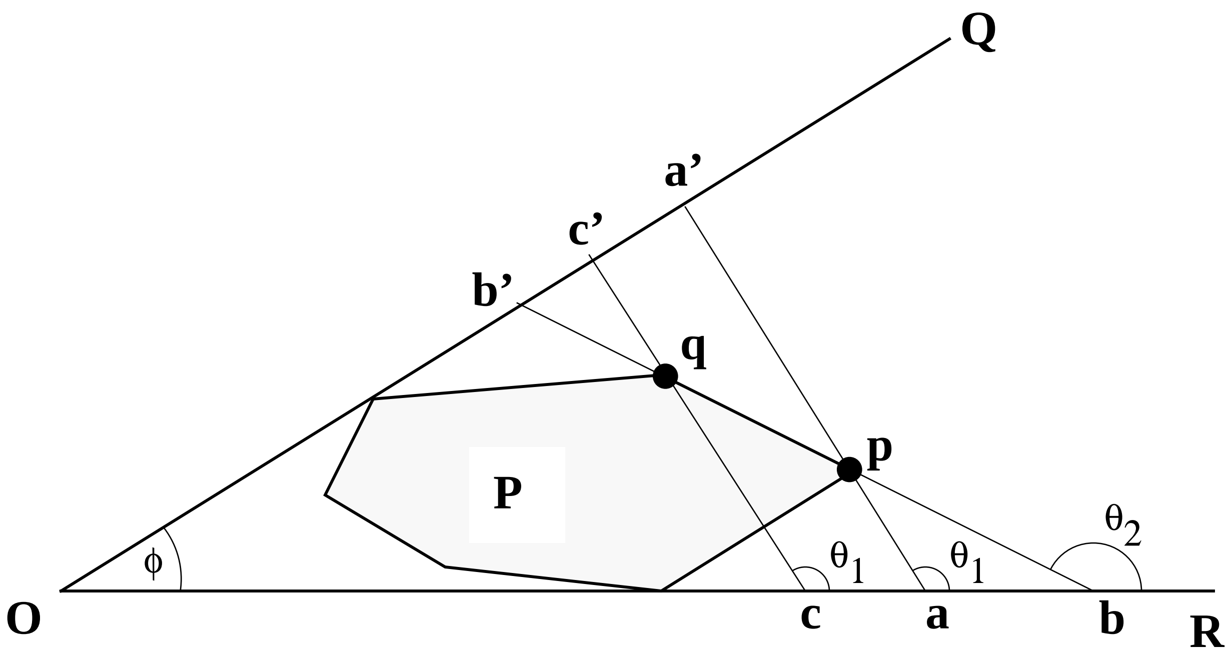

Fig. 2.4: Illustrating the proof of lemma 2.2: case 1: l(aa') is increasing at $\theta = \theta_1$.

$< \theta_2$ or it first decreases to a minimal length and then increases. In either case it follows from lemma 2.1 that l(HK(q,θ)) continues to increase for $\theta_2 < \theta < \theta_3$, where θ_3 is either π or the angle determined by HK(qr, θ_3), as the case may be, where r is the vertex of P adjacent to q in a counterclockwise order about *bd*(P). Similar arguments apply to succeeding vertices of P. Therefore we conclude that if l(HK(P,θ)) is increasing at θ_1 it continues to increase for $\theta_1 < \theta < \pi$.

Case 2: l(aa') is decreasing at $\theta = \theta_1$. (Fig. 2.5)

Let p, q, aa', and bb' be as in *Case 1*. Consider HK(p,θ) for $\theta_1 < \theta < \theta_2$. From the unimodality of l(HK(p,θ)) by lemma 2.1 it follows that we have two subcases.

Subcase 2.1: l(HK(p,θ)) decreases continually until it reaches a minimal length $l^* = l$(HK(p,θ^*)) at $\theta = \theta^*$ and subsequently increases continually for $\theta^* < \theta < \theta_2$. In this subcase for $\theta^* < \theta < \theta_2$ we have an instance of *Case 1* and l(HK(P,θ)) will continue to increase for $\theta > \theta_2$. Note that in this subcase l^* is in fact the minimum of l(HK(P,θ)) in the entire interval $\phi < \theta < \pi$.

Subcase 2.2: l(HK(p,θ)) decreases continually in the interval $\theta_1 < \theta < \theta_2$. Construct HK(q,$\theta$) such that l(HK(q,θ)) is the minimum over $\phi < \theta < \pi$. Denote such an HK by HK*. Two cases arise depending on whether or not p lies in the interior of ΔOHK.

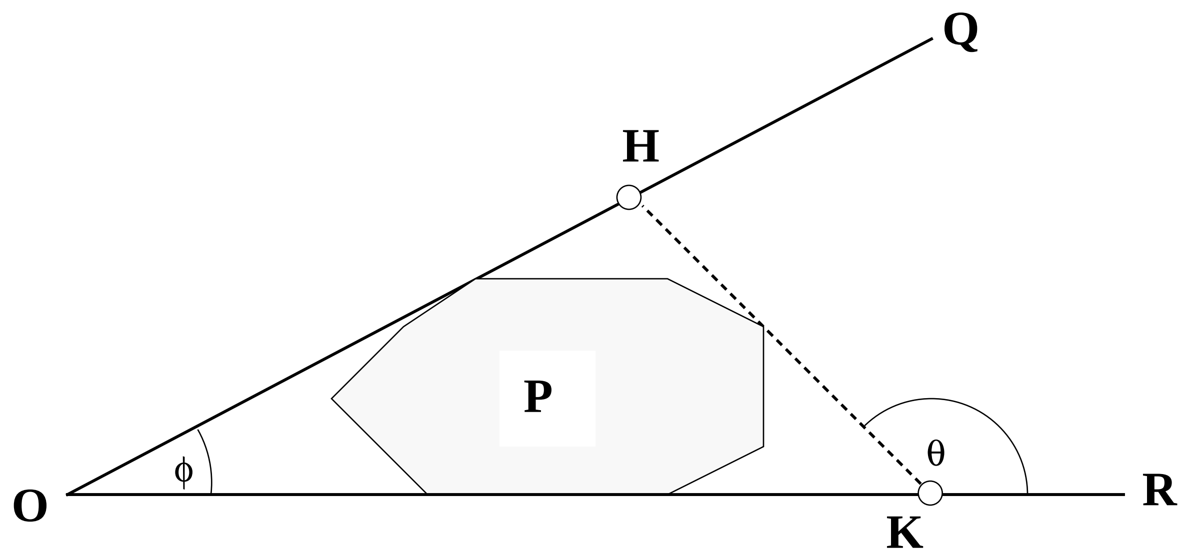

Fig. 2.3: Let P be a convex polygon contained in QOR. Let H slide along OQ and K slide along OR. Then, as θ increases from ϕ to π with the constraint that HK remain tangent P , the length of HK is a *unimodal* function of θ.

then HK may traverse the interior of P. Similarly, HK(pq,θ) denotes such a line segment constrained to be colinear with line segment pq. Let HK(P,θ) denote the line segment HK as above such that HK is tangent to polygon P and P is contained in ΔOHK. Finally, let *l*(HK(.)) denote the length of the line segment HK(.).

Lemma 2.2: For $\phi < \theta < \pi$ we have that *l*(HK(P,θ)) is a *unimodal* function of θ.

Proof: Consider any fixed position for HK(P,θ). We will show that *l*(HK(P,θ)) is either continually increasing as a function of θ or it is first continually decreasing to a minimal length *l**, attained at some angle θ^*, and subsequently continually increasing for $\theta > \theta^*$. Therefore let the starting position of HK(P,θ) be HK(p,θ_1) = aa’ and refer to Fig. 2.4. We have two cases: either *l*(aa’) is increasing at $\theta = \theta_1$ or it is decreasing.

Case 1: *l*(aa’) is increasing at $\theta = \theta_1$. (Fig. 2.4)

Let q be the vertex of P adjacent to p in counter-clockwise order about *bd*(P). Construct HK(q,θ_1) = cc’ and HK(pq,θ_2) = bb’. Since cc’ is parallel to aa’ and closer to O than aa’, it follows that *l*(cc’) < *l*(aa’). Since *l*(aa’) is increasing at $\theta = \theta_1$ and *l*(aa’) = *l*(HK(p,θ_1)) is unimodal by lemma 2.1, we have that *l*(aa’) < *l*(bb’). Therefore *l*(cc’) < *l*(bb’). Now consider HK(q,θ) as θ varies from θ_1 to θ_2. From lemma 2.1 it follows that *l*(HK(q,θ)) is unimodal and therefore we must have two possible situations. Either *l*(HK(q,θ)) continually increases in the interval $\theta_1 < \theta$

< 0 is for either all three roots to be positive or one to be positive and the other two negative. In addition since $b < 0$ we must have $r_1r_2 + r_2r_3 + r_3r_1 < 0$. Thus consider the case in which all three roots are positive. This implies $r_1r_2 + r_2r_3 + r_3r_1 > 0$, a contradiction. It follows that one root must be positive and the other two negative. We can in a similar way show that when only one root of (2) is real that root has to be a positive one. For assume that we have, as before,

$$x^3 + ax^2 + bx + c = 0 \text{ with } a,b,c < 0,$$

and that we have two complex roots and one real. Let r_1 be the real root. Then we can rewrite this equation as

$$(x-r_1)(x^2 + ex + f) = 0.$$

For there to be two complex roots we must have $e^2-4f < 0$, which implies in turn that $f > 0$.

Since $c=-r_1f < 0$ and $f > 0$ it follows that $r_1 > 0$.

We conclude from the above discussion that we need only be concerned with the positive root. Let $x = r$ be the positive root of (2). We now show that $l(x)$ attains a minimum value at $x = r$. We do this by demonstrating that the second derivative of $l(x)$ evaluated at $x = r$ must be greater than zero.

Let $A = x^3-2ux^2+(u^2-au)x-a\beta^2$. Then $l'(x) = (\beta^2 + x^2)^{-1/2} (x-u)^{-2} A$. It is sufficient to show that $dA/dx > 0$ at $x = r$.

We see that
$$dA/dx = 3x^2 - 4ux + u^2 - au \qquad (4)$$

Evaluating (4) at $x = r$ we obtain

$$\begin{aligned} dA/dx(r) &= 3r^2 - 4ur + u^2 - au \\ &= (2r^2 - 2ur) + (r^2 - 2ur + (u^2 - au)) \\ &= (2r^2 - 2ur) + a\beta^2/r \text{ , since } r^3-2ur^2+(u^2-au)r-a\beta^2 = 0. \\ &= 2r(r-u) + a\beta^2/r \text{ .} \qquad (5) \end{aligned}$$

Since all terms in (5) are positive and $r > u$, dA/dx at $x = r > 0$, and $l(x)$ attains its minimum value at $x = r$. Since this minimum is the only local minimum in the region of interest, unimodality in the region of interest follows. Q.E.D.

In order to obtain efficient algorithms for our problems we actually need a more general result than that provided by lemma 2.1. We need an analogous unimodality result for the case when point p is actually a convex polygon P and the line segment HK is constrained to remain tangential to P as θ increases from ϕ to π as illustrated in Fig. 2.3. Accordingly, let $HK(p,\theta)$ denote the line segment HK with end point H on OQ and endpoint K on OR such that HK traverses point p and makes an angle θ with respect to OR as an axis of reference (see Fig. 2.3). Note that if p is a vertex of P

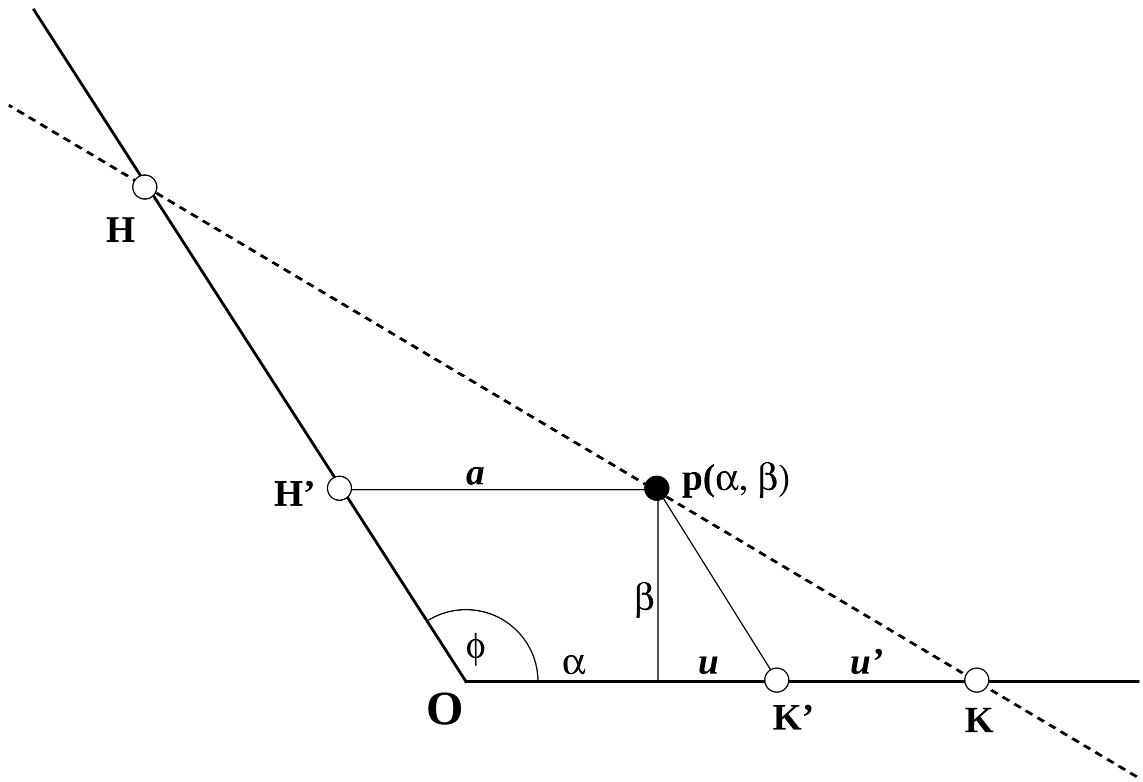

Fig. 2.2: Illustrating the proof of lemma 2.1. O represents the origin and x=u+u'.

Since $x > u$ it follows that $l'(x) = 0$ implies that

$$x^3 - 2ux^2 + (u^2 - au)x - a\beta^2 = 0 \qquad (2)$$

Rewriting (2) in standard form we have that

$$x^3 + ax^2 + bx + c = 0$$

where $a = -2u < 0$; $b = u(u-a) < 0$, since $a > u$, and $c = -a\beta^2 < 0$.

Let r_1, r_2, and r_3 be the roots of (2). We assume that r_1, r_2, and r_3 are all real. Then we can write

$$x^3 + ax^2 + bx + c = (x-r_1)(x-r_2)(x-r_3) \,. \qquad (3)$$

We now show that we need only examine one root in more detail. Since a, b, and c, are all less than zero we argue that precisely one of the roots must be greater than zero and the other two must be less than zero. We start by rewriting (3) as follows:

$$x^3 + ax^2 + bx + c = x^3 - (r_1 + r_2 + r_3)x^2 + (r_1r_2 + r_2r_3 + r_3r_1)x - r_1r_2r_3 = 0$$

where $a = -(r_1 + r_2 + r_3)$, $b = (r_1r_2 + r_2r_3 + r_3r_1)$ and $c = -r_1r_2r_3$. Now $c < 0$ and the only way $-r_1r_2r_3$

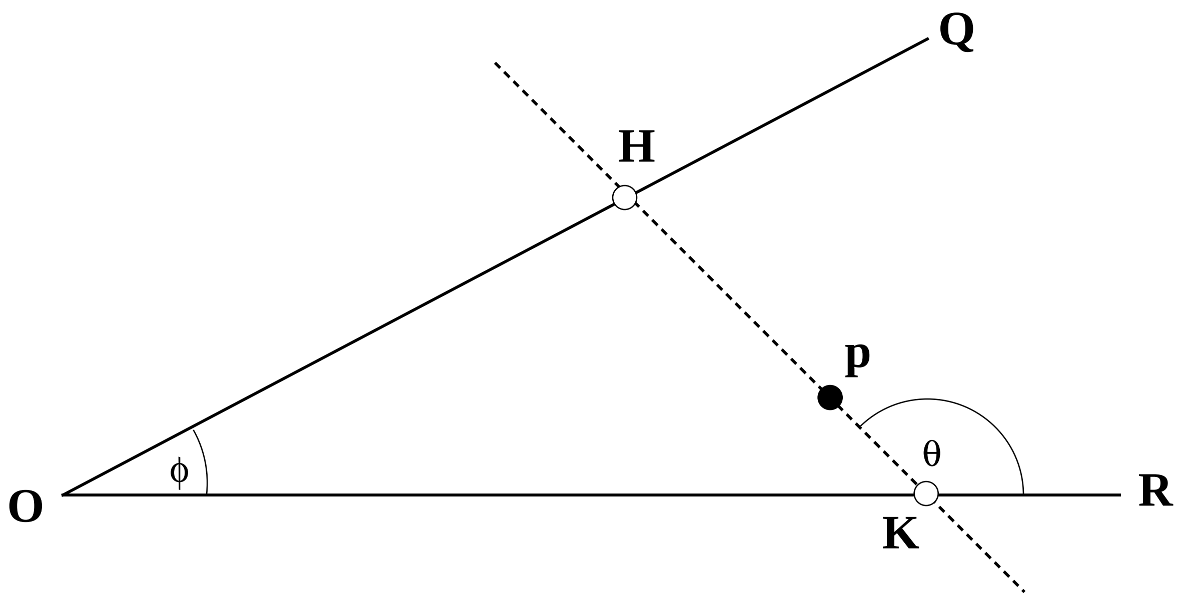

Fig. 2.1: Illustrating the fundamental geometric minimization problem.

In order to simplify analysis it is convenient to break up the problem into two cases depending on whether ϕ is greater than $\pi/2$ or not, and to parameterize the problem not as a function of θ as in Fig. 2.1, but rather as a function of $x=u+u'$ as in Fig. 2.2. It is straightforward to demostrate that unimodality under the first parameterization implies unimodality under the second. Furthermore, we restrict our analysis to the more difficult case of ϕ greater than $\pi/2$. The analysis for ϕ less than or equal to $\pi/2$ is similar and less involved. Accordingly, let $p(\alpha,\beta)$ be the point in the interior of the cone in question. Thus $\beta > 0$. H'p has length a and is parallel to OK. K'p is parallel to OH. Let $l(x)$ denote the length of line segment HK as a function of x. We then have the following lemma.

Lemma 2.1: For $x > u > 0$, $l(x)$ is a *unimodal* function.

Proof: Without loss of generality we assume that $\alpha > 0$. For otherwise we may construct a symmetrical diagram where β represents the perpendicular drop from p to OH rather than OK. First we determine the value of x such that HK is a minimum.

Since ΔpHH' and ΔpK'K are similar it follows that $pH/a = pK/(x-u)$ and thus, $(pH+pK)/pK = 1+a/(x-u)$. Since $l(x)=pH+pK$ we can write $l(x)$ as follows:

$$l(x) = (\beta^2 + x^2)^{1/2} (1 + a/(x-u)) . \quad (1)$$

Differentiating (1) with respect to x we obtain,

$$l'(x) = (\beta^2 + x^2)^{-1/2} (x-u)^{-2} [x^3-2ux^2+(u^2-au)x-a\beta^2] .$$

erations as their intersection, common tangents, etc. O(n log n) time is sufficient for the special cases of unsorted vertical line segments [O'R81], for line segments with arbitrary directions [EMPRWW82], for a set of n translates of a simple object in the plane [Ed85] and for n circles of equal radius [BL83]. Given a family *K* of n convex cones, determining whether *K* admits a common transversal can be accomplished in O(n log n α(n)) time, where α(n) is the extremely slowly growing inverse Ackermann's function, with the technique of Atallah and Bajaj [AB87]. This result can now be improved however with the O(n log n) time algorithm of Hershberger [He89] for finding the upper envelope of n line segments. If on the other hand each cone in *K* is a maximal unobstructed external-visibility cone anchored at a vertex of a simple polygon then this additional structure can be exploited to determine if a common transversal exists in O(n) time [BKT89]. Finally we mention that high dimensional transversal problems have been recently investigated by Avis & Doskas [AD90] and [Ro88].

In this paper we investigate the hitherto unexplored problem of computing the *shortest* transversals when they exist. We present an O(n $\log^2$ n) time and O(n) space algorithm for computing the shortest transversal of a set of n given line segments or lines in the plane. The length of such a transversal can be viewed as a measure of the *girth* of an *arrangement.* If the line segments do not intersect the algorithm can be trimmed to run in O(n log n) time. Furthermore, in conjunction with convex hull and linear programming components the algorithm will also find the shortest line segment that intersects a set of n isothetic rectangles in O(n log k) time, where k is the combinatorial complexity of the space of transversals and $k \leq 4n$. These results find application in: (1) line-fitting between a set of n data ranges where it is desired to obtain the shortest *line-of-fit*, (2) finding the shortest line segment from which a convex n-vertex polygon is weakly externally visible, and (3) determining the shortest *line-of-sight* between two edges of a simple n-vertex polygon, for which O(n) time algorithms are also given All the algorithms are based on the solution to a fundamental geometric minimization problem that is of independent interest and should find application in several different contexts.

2. A Fundamental Geometric Optimization Problem

The foundation of our algorithm consists of a geometric minimization problem that, surprisingly, appears not to have been investigated by geometers in the past. Let OQ and OR denote two infinite half rays emanating at O and subtending an angle ϕ as OR is rotated about O in a counterclockwise direction until it coincides with OQ, and refer to Fig. 2.1. Note that Q and R are marked for convenience but should be interepreted as lying at infinity. Thus ROQ denotes an unbounded cone. Let p be any point in the interior of this cone. We would like to find the shortest line segment [H,K] such that H lies on OQ and K lies on OR and p lies on [H,K]. In other words we require the shortest straight-line path that will connect OQ with OR with the constraint that the path traverse the point p. In actual fact we require more than that in order to design our algorithm. Let θ be the angle subtended by RKH as in Fig. 2.1 and let *l* denote the length of line segment [H,K]. Let $l(\theta)$ denote *l* as a function of θ where $\phi = \theta_1 < \theta \leq \theta_2 < \pi$. We require that $l(\theta)$ be a *unimodal* function of θ for the interval of interest. We conjecture the stronger result that $l(\theta)$ is in fact a *convex* function but we leave this as an open problem.

Computing Shortest Transversals

Binay Bhattacharya
School of Computing Science
Simon Fraser University
Burnaby

and

Godfried Toussaint
School of Computer Science
McGill University
Montreal

ABSTRACT

We present an O(n log^2 n) time and O(n) space algorithm for computing the shortest line segment that intersects a set of n given line segments or lines in the plane. If the line segments do not intersect the algorithm may be trimmed to run in O(n log n) time. Furthermore, in combination with linear programming the algorithm will also find the shortest line segment that intersects a set of n isothetic rectangles in the plane in O(n log k) time, where k is the combinatorial complexity of the space of transversals and $k \leq 4n$. These results find application in: (1) line-fitting between a set of n data ranges where it is desired to obtain the shortest *line-of-fit*, (2) finding the shortest line segment from which a convex n-vertex polygon is weakly externally visible, and (3) determining the shortest *line-of-sight* between two edges of a simple n-vertex polygon, for which O(n) time algorithms are also given. All the algorithms are based on the solution to a new fundamental geometric optimization problem that is of independent interest and should find application in different contexts as well.

1. Introduction

Common transversals for families of convex sets have been investigated for some time in both the mathematics [Gr58], [Le80] and computer science [AB87], [AW87], [AW88], [Ed85], [We88] literatures. In the computer science literature the more aggressive term *stabber* is traditionally used for transversal. Transversals in the plane find application in several areas including line-fitting [O'R81] and updating triangulations [ET85]. Edelsbrunner, Overmars and Wood [EOW81] developed a method for solving planar visibility problems that yields a procedure for computing transversals for F, a family of simple objects, in O(n^2 log n) time, where n is the cardinality of F. By simple objects it is meant those objects that have an O(1) storage description each and which are such that, for every pair of objects, constant time suffices to compute such basic op-