Purdue University

Purdue e-Pubs

Department of Computer Science Technical Reports

Department of Computer Science

1992

Optimally Computing the Shortest Weakly Visible Subedge for a Simple Polygon

Danny Z. Chen

Report Number:

92-028

OPTIMALLY COMPUTING THE SHORTEST WEAKLY VISIBLE SUBEDGE OF A SIMPLE POLYGON

Danny Z. Chen

CSD-TR-92-028
May 1992

Optimally Computing the Shortest Weakly Visible Subedge of a Simple Polygon*

Danny Z. Chen
Department of Computer Science and Engineering
University of Notre Dame
Notre Dame, IN 46556
chen@cse.nd.edu

Abstract

Given an n-vertex simple polygon P, the problem of computing the shortest weakly visible subedge of P is that of finding a shortest line segment s on the boundary of P such that P is weakly visible from s (if s exists). In this paper, we present new geometric observations that are useful for solving this problem. Based on these geometric observations, we obtain optimal sequential and parallel algorithms for solving this problem. Our sequential algorithm runs in $O(n)$ time, and our parallel algorithm runs in $O(\log n)$ time using $O(n/\log n)$ processors in the CREW PRAM computational model. Using the previously best known sequential algorithms to solve this problem would take $O(n^2)$ time. We also give geometric observations that lead to extremely simple and optimal algorithms for solving, both sequentially and in parallel, the case of this problem where the polygons are rectilinear.

*This research was partially done when the author was with the Department of Computer Sciences, Purdue University, West Lafayette, Indiana, and was supported in part by the Office of Naval Research under Grants N00014-84-K-0502 and N00014-86-K-0689, the National Science Foundation under Grant DCR-8451393, and the National Library of Medicine under Grant R01-LM05118.

1 Introduction

Given a set of "opaque" geometric objects, two points p and q are said to be *visible* from each other iff the interior of the line segment $\overline{pq}$ does not intersect any of these opaque objects. Visibility is one of the most fundamental topics in computational geometry. Visibility problems find applications in many areas, such as computer graphics, computer vision, VLSI design, and robotics. Visibility problems also appear as subproblems in other geometric problems (like finding the shortest obstacle-avoiding paths and computing intersections between geometric figures). Numerous efficient algorithms have been designed for solving various visibility problems, in both sequential and parallel computational models.

In this paper, we consider a *weak visibility* problem. Weak visibility deals with visibility problems in which the "observers" are of the shape of line segments. An important class of weak visibility problems studies the case where the opaque objects are the boundaries of simple polygons. For a point p in a polygon P and a line segment s, p is said to be *weakly visible* from s iff p is visible from some point on s (depending on p). Polygon P is said to be *weakly visible* from a line segment s iff every point $p \in P$ is weakly visible from s. Many sequential algorithms [1, 2, 3, 4, 7, 8, 9, 10, 12, 13, 15, 18, 19, 20, 21, 22, 23] and parallel algorithms [5, 6, 11, 14] for solving various weak visibility problems on simple polygons have been discovered.

We consider the problem of computing the shortest weakly visible subedge of a simple polygon (called it the SWVS problem). That is, given an n-vertex simple polygon P, we would like to find a line segment s on the boundary of P such that (i) P is weakly visible from s (if s exists), and (ii) the length of s is the shortest among all such line segments on the boundary of P (it is possible that s is a single point on the boundary of P). Intuitively, if P represents a house whose interior is that of a simple polygon, then s is the shortest portion of any wall of P by which a guard has to patrol back and forth in order to keep the inside of P completely under surveillance.

There is related work on the SWVS problem. Avis and Toussaint [1] considered the problem of detecting the weak visibility of a simple polygon (that is, deciding whether a polygon P is weakly visible from an edge e of P, and reporting all such edges e for P); they presented a sequential linear time algorithm for the case of checking whether P is weakly visible from a *specified* edge e of P. Another sequential linear time algorithm for this case was given in [10]. Sack and Suri [20] and Shin [21] independently gave optimal linear time algorithms for solving the problem of detecting the weak visibility of a simple polygon. Chen [5] came up with an optimal parallel algorithm for this problem; Chen's algorithm runs in $O(\log n)$ time using $O(n/\log n)$ CREW PRAM processors.

Several problems on computing weakly visible line segments with respect to a simple polygon have been studied. Ke [15] and Doh and Chwa [8] gave $O(n \log n)$ time algorithms for computing a line segment in a polygon from which the polygon is weakly visible (such a segment can be in the interior of the polygon); in particular, Ke's algorithm finds such a line segment of shortest length. Lee and Chwa [19] designed a linear time algorithm for computing all the maximal convex chains or all the maximal reflex chains on the boundary of a polygon from which the polygon is weakly visible. Bhattacharya *et al.* [3] presented a linear time algorithm for computing a shortest line segment (not in the interior of a polygon) from which the boundary of the polygon is weakly visible (or *externally visible*). Ching *et al.* [7] showed that, if a polygon is weakly visible from a *specified* edge e, then the shortest weakly visible subedge on e can be computed in linear time by using the algorithm in [1]. The problem of computing in parallel the shortest weakly visible subedge *on a specified polygon edge* was solved optimally by Chen [6], in $O(\log n)$ time using $O(n/\log n)$ CREW PRAM processors.

The SWVS problem, obviously, is a natural generalization of the weak visibility problem first studied by Avis and Toussaint [1] and then by Sack and Suri [20] and Shin [21]. A straightforward sequential solution to the SWVS problem based on these known algorithms would consist of the following steps: (1) Compute all the edges of P from each of which P is weakly visible, by using [20, 21]. (2) For every edge so obtained, compute the shortest weakly visible subedge on that edge, by using [1, 7]. (3) Among all the weakly visible subedges computed in step (2), find the one with the shortest length. Such an algorithm certainly solves the SWVS problem correctly. However, because a simple polygon can have $O(n)$ edges from each of which the polygon is weakly visible, and because computing the shortest weakly visible subedge on a specified edge in general requires $O(n)$ time [1, 7], the above algorithm takes $O(n^2)$ time.

In this paper, we present new geometric observations that are useful for solving the SWVS problem. Based on these geometric observations, we obtain efficient sequential and parallel algorithms for solving the SWVS problem. Our sequential algorithm runs in $O(n)$ time, and our parallel algorithm runs in $O(\log n)$ time using $O(n/\log n)$ CREW PRAM processors. These algorithms are obviously optimal. We also give geometric observations that lead to extremely simple and optimal algorithms for solving, both sequentially and in parallel, the case of the SWVS problem where the polygons are rectilinear (i.e., the edges of the polygons are either vertical or horizontal).

The parallel computational model we use is the CREW PRAM; this is the synchronous shared-memory model where multiple processors can simultaneously read from the same memory location but at most one processor is allowed to write to a memory location at each time unit. We also use the EREW PRAM model, in which no simultaneous accesses to the same memory location are allowed.

The rest of this paper consists of 5 sections. Section 2 gives some notation and preliminary results needed in the paper. Section 3 presents the crucial geometric observations used by our algo-

Figure 1: The weakly visible edges of P are e_1, e_2, and e_4.

rithms. Section 4 describes the sequential and parallel algorithms for solving the SWVS problem. Section 5 gives the simple algorithms for the case of the SWVS problem on rectilinear polygons. Section 6 concludes the paper.

2 Preliminaries

The input to the SWVS problem consists of an n-vertex simple polygon P, and the output is s, the shortest weakly visible subedge of P (if s exists). Polygon P is specified by a sequence (v_1, v_2, $\ldots$, v_n) of its vertices, in the order in which they are visited by a *counterclockwise* walk along the boundary of P, starting from vertex v_1. Without loss of generality (WLOG), we assume that no three vertices of P are collinear.

The edge of P joining v_i and v_{i+1} is denoted by $e_i = \overline{v_i v_{i+1}}$ ($= \overline{v_{i+1} v_i}$), with the convention that $v_{n+1} = v_1$. The boundary of P is denoted by $bd(P)$, and the polygonal chain from v_i counterclockwise to v_j along $bd(P)$ is denoted by $C(i, j)$. The size of a chain C is the number of line segments on C, denoted by $|C|$.

An edge e of P from which P is weakly visible is called a *weakly visible edge* of P. We denote the set of all the weakly visible edges of P by WVE. In Figure 1, for example, $WVE = \{e_1, e_2, e_4\}$. Note that, for an arbitrary simple polygon of n vertices, the set of its weakly visible edges can be computed optimally, in $O(n)$ time sequentially [20, 21], and in $O(\log n)$ time using $O(n/\log n)$ CREW PRAM processors in parallel [5]. WLOG, we assume that $WVE \neq \emptyset$ (because if $WVE = \emptyset$, then P is not weakly visible from any of its edges and hence the shortest weakly visible subedge s on $bd(P)$ does not exist). For each edge $e \in WVE$, we denote the shortest weakly visible subedge of P on e by $s(e)$.

Let $WVE = \{we_1, we_2, \ldots, we_m\}$, where $m = |WVE|$. Note that m can be $O(n)$. WLOG, we assume that $m > c$ for some constant integer $c \geq 1$ (c will be decided in Section 3). This is because if $m \leq c$, then s is one of the $m = O(1)$ $s(e)$'s, where $e \in WVE$. The $O(1)$ $s(e)$'s can be computed optimally, both sequentially and in parallel, by respectively applying the algorithms in [1, 6] to every edge $e \in WVE$.

We label the we_i's of WVE in such a way that $we_1 = e_1$ and that, when walking along $bd(P)$ counterclockwise by starting at v_1, we visit the we_i's in increasing order of their indices. In the rest of this paper, we use the following convention for the indices of the we_i's: For every integer $i = 1, 2, \ldots, m$, $we_{i+m} = we_i$, and for every integer $j = 0, 1, \ldots, m-1$, $we_{-j} = we_{m-j}$.

For an edge $we_i = e_j \in WVE$, we call v_j (resp., v_{j+1}) the *first vertex* (resp., *last vertex*) of we_i, and denote it by $fv(we_i)$ (resp., $lv(we_i)$). For two consecutive edges we_i and we_{i+1} of WVE, where $we_i = e_j$ and $we_{i+1} = e_k$, we denote by C_i the chain on $bd(P)$ from $lv(we_i)$ counterclockwise to $fv(we_{i+1})$ *excluding* $lv(we_i)$ and $fv(we_{i+1})$. Note that $C_i = (e_{j+1}, e_{j+2}, \ldots, e_{k-1}) - \{v_{j+1}, v_k\}$, and that C_i contains no edge in WVE. C_i can be $\emptyset$ for some i (when $lv(we_i) = fv(we_{i+1})$). Obviously, the we_i's and C_i's together form a partition of $bd(P)$.

A point p in the plane is represented by its x-coordinate and y-coordinate, denoted by $x(p)$ and $y(p)$, respectively. For three non-collinear points p, q, and r, we say that the directed chain from p to q to r makes a *left* (resp., *right*) turn iff $x(r)(y(p)-y(q)) + y(r)(x(q)-x(p)) + x(p)y(q)-x(q)y(p) > 0$ (resp., < 0). For a directed simple chain $C = (p_1, p_2, \ldots, p_k)$, $k \geq 3$, C is said to make only left (resp., right) turns iff every subchain of the form (p_{i-1}, p_i, p_{i+1}) makes a left (resp., right) turn, $1 < i < k$.

A vertex v_i is *convex* if the interior angle of P at v_i is $< \pi$. An edge e_i is *convex* if both v_i and v_{i+1} are convex. For any edge $we_i \in WVE$, if we_i is convex, then for any subchain $C(j,k)$ of $C(lv(we_i), fv(we_i))$, the (directed) shortest path from v_j to v_k inside P goes through only the vertices on $C(j,k)$, and the shortest path makes only right turns (this fact is shown in [1, 10]). Hence, we call such a shortest path the *internal convex path* between v_j and v_k along $C(j,k)$.

3 Useful Geometric Observations

In this section, we present useful geometric observations for solving the SWVS problem. The observations that we give here are new. It is these geometric observations that enable us to achieve the optimal algorithms to be given in the next section.

The idea of our algorithms is to compute the shortest weakly visible subedge $s(we_i)$ on every edge $we_i \in WVE$. Because $|WVE|$ can be $O(n)$ and because computing each $s(we_i)$ in general requires $O(n)$ operations, the algorithms based on this idea appear to take $O(n^2)$ operations. The following lemmas are crucial to the optimality of our algorithms.

Lemma 1 *Suppose that $|WVE| \geq 7$. Then for every edge $we_i \in WVE$, the following are true:*

(1) *The vertex $fv(we_i)$ is visible from every point on the chain along $bd(P)$ from vertex u'* clockwise

Figure 2: The view between $fv(we_i)$ and p cannot be blocked: **Case (i)**.

Figure 3: The view between $fv(we_i)$ and p cannot be blocked: **Case (ii)**.

to vertex v', where $u' = fv(we_{i-2})$ if $C_{i-2} \neq \emptyset$ and $u' = fv(we_{i-3})$ otherwise, and $v' = lv(we_{i+1})$ if $C_i \neq \emptyset$ and $v' = lv(we_{i+2})$ otherwise.

(2) *The vertex $lv(we_i)$ is visible from every point on the chain along $bd(P)$ from vertex u''* counterclockwise *to vertex v'', where $u'' = lv(we_{i+2})$ if $C_{i+1} \neq \emptyset$ and $u'' = lv(we_{i+3})$ otherwise, and $v'' = fv(we_{i-1})$ if $C_{i-1} \neq \emptyset$ and $v'' = fv(we_{i-2})$ otherwise.*

Proof. Note that, because $|WVE| \geq 7$, the chains defined in (1) and (2) both do not contain we_i. We only prove (1) (the proof for (2) is symmetric).

We first prove the case where C_{i-2} and C_i are both nonempty. Let p be an arbitrary point on the chain along $bd(P)$ from $fv(we_{i-2})$ clockwise to $lv(we_{i+1})$. To prove that p is visible from $fv(we_i)$, we need to show that the following are true: (i) The chain along $bd(P)$ from $fv(we_i)$ *clockwise* to p does not block the view between $fv(we_i)$ and p, and (ii) the chain along $bd(P)$ from $fv(we_i)$ *counterclockwise* to p does not block the view between $fv(we_i)$ and p.

Case (i) Let q be a point on C_{i-2}. If the view between $fv(we_i)$ and p were blocked by the chain along $bd(P)$ from q counterclockwise to $fv(we_i)$, then $fv(we_i)$ would have not been weakly

Figure 4: The view between $fv(we_i)$ and $lv(we_{i+2})$ cannot be blocked: When $C_i = \emptyset$.

visible from we_{i-2} (see Figure 2 (a)), a contradiction. If the view between $fv(we_i)$ and p were blocked by the chain along $bd(P)$ from p counterclockwise to q, then p would have not been weakly visible from we_{i-1} (see Figure 2 (b)), again a contradiction.

Case (ii) Let q be a point on C_i. If the view between $fv(we_i)$ and p were blocked by the chain along $bd(P)$ from $fv(we_i)$ counterclockwise to q, then $fv(we_i)$ would have not been weakly visible from we_{i+1} (see Figure 3 (a)), a contradiction. If the view between $fv(we_i)$ and p were blocked by the chain along $bd(P)$ from q counterclockwise to p, then p would have not been weakly visible from we_i (see Figure 3 (b)), again a contradiction.

Suppose that $C_i = \emptyset$. We need to show that the chain along $bd(P)$ from $fv(we_i)$ counterclockwise to $lv(we_{i+2})$ does not block the view between $fv(we_i)$ and $lv(we_{i+2})$. If the view were blocked by the chain along $bd(P)$ from $fv(we_i)$ counterclockwise to $fv(we_{i+2})$ excluding $fv(we_{i+2})$, then $fv(we_i)$ would have not been weakly visible from we_{i+2} (see Figure 4 (a)), a contradiction. If the view were blocked by we_{i+2} itself, then $lv(we_{i+2})$ would have not been weakly visible from we_i (see Figure 4 (b)), again a contradiction. The proof for other points on the chain along $bd(P)$ from $fv(we_{i-2})$ clockwise to $lv(we_{i+2})$ is similar to the proof of Cases (i) and (ii) above (with edge we_{i+1} playing the role of C_i).

The case where $C_{i-2} = \emptyset$ is also proved similarly to Cases (i) and (ii). This is because the chain along $bd(P)$ from $lv(we_{i-3})$ counterclockwise to $fv(we_{i-1})$ is nonempty, and hence it can play the role of C_{i-2} in the above proof. For an example of $fv(we_{i-2})$ not visible from $fv(we_i)$ when $C_{i-2} = \emptyset$, see Figure 5. $\square$

Lemma 2 *Suppose that $|WVE| \geq 7$. Then for each edge $we_i \in WVE$, we_i is completely visible from every point on the chain along $bd(P)$ from vertex u* clockwise *to vertex v, where $u = fv(we_{i-2})$ if $C_{i-2} \neq \emptyset$ and $u = fv(we_{i-3})$ otherwise, and $v = lv(we_{i+2})$ if $C_{i+1} \neq \emptyset$ and $v = lv(we_{i+3})$ otherwise.*

Figure 5: Illustrating the situation where $fv(we_{i-2})$ is not visible from $fv(we_i)$.

Figure 6: Illustrating the proof of Lemma 2.

Proof. Let C^i_{vu} be the chain along $bd(P)$ from u clockwise to v. Because $|WVE| \geq 7$, C^i_{vu} does not contain we_i. By Lemma 1, every point p on C^i_{vu} is visible from both endpoints $fv(we_i)$ and $lv(we_i)$ of we_i. Hence it is easy to see that p is visible from every point on we_i (see Figure 6). □

For every $we_i \in WVE$, let C^i_{vu} denote the chain along $bd(P)$ from vertex u clockwise to vertex v as defined in Lemma 2. The computational consequence of Lemma 2 is that, when computing $s(we_i)$ on every edge $we_i \in WVE$, we can simply ignore the effect of all the points on C^i_{vu}. This is because, by Lemma 2, edge we_i is completely visible from every point on C^i_{vu}. The points in P that we need to consider when computing $s(we_i)$, therefore, are all on the following two disjoint subchains of $bd(P)$:

(a) The chain from u counterclockwise to $fv(we_i)$, denoted by LC_i, and

(b) the chain from v clockwise to $lv(we_i)$, denoted by RC_i.

In summary, for every $we_i \in WVE$, the computation of $s(we_i)$ is based only on chains LC_i and RC_i.

Figure 7: Illustrating the proof of Lemma 3.

Note that chain LC_i contains at most two nonempty chains C_j, where $j \in \{i-1, i-2, i-3\}$, and that RC_i contains at most two nonempty chains C_k, where $k \in \{i, i+1, i+2\}$. We only discuss the computation of $s(we_i)$ with respect to the points on RC_i (the computation of $s(we_i)$ with respect to LC_i is similar).

WLOG, we assume for the rest of this section that $|WVE| \geq 7$. Note that, based on the lemmas in this section, the integer parameter c of our algorithms (c was introduced in Section 2) is chosen to be 7.

The next lemma greatly reduces our effort in computing $s(we_i)$ with respect to the points on chain RC_i: It enables us to "localize" the computation to RC_i.

Lemma 3 *For every point p on RC_i and every point q on we_i, the chain along $bd(P)$ from p* counterclockwise *to q does not block the view between p and q.*

Proof. Suppose that the chain C_{pq} along $bd(P)$ from p *counterclockwise* to q did block the view between p and q. Let $ICP(C_{pq})$ be the internal convex path between p and q that passes *only* the vertices of C_{pq}, and let $\overline{pq'}$ be the line segment on $ICP(C_{pq})$ that is adjacent to p (see Figure 7). Since $|WVE| \geq 7$, there must be at least one edge $we_j \in WVE$ such that (1) we_j is not adjacent to q', and (2) we_j is either on the subchain of C_{pq} from p counterclockwise to q' or on the subchain of C_{pq} from q' counterclockwise to q. If we_j is on the subchain of C_{pq} from p counterclockwise to q', then q would have not been weakly visible from we_j, a contradiction. If we_j is on the subchain of C_{pq} from q' counterclockwise to q, then p would have not been weakly visible from we_j, again a contradiction. □

By Lemma 3, for every point p on RC_i and every point q on we_i, the view between p and q can be blocked only by the chain along $bd(P)$ from q counterclockwise to p.

We now consider the computation of $s(we_i)$ with respect to the points on RC_i. We further partition RC_i into two subchains: (a) The chain from the endpoint v of RC_i (as defined in Lemma 2) *clockwise* to $lv(we_{i+1})$ excluding $lv(we_{i+1})$, denoted by RC_i^r, and (b) the chain from $lv(we_{i+1})$ *clockwise* to $lv(we_i)$, denoted by RC_i^l. The following lemmas are useful in computing $s(we_i)$.

Figure 8: Illustrating Lemma 5 (with $we_j = we_{i+1}$).

Lemma 4 *For every point p on RC_i, if $fv(we_i)$ is not visible from p, then $lv(we_i)$ must be visible from p.*

Proof. By Lemma 3, the view between p and $fv(we_i)$ cannot be blocked by the chain along $bd(P)$ from p counterclockwise to $fv(we_i)$. So if $fv(we_i)$ is not visible from p, the view must be blocked by the chain C' along $bd(P)$ from p *clockwise* to $fv(we_i)$. But if the view between p and $lv(we_i)$ were also blocked by C', then p would have not been weakly visible from we_i, a contradiction. □

Corollary 1 *For every point p on RC_i, if $lv(we_i)$ is not visible from p, then $fv(we_i)$ is visible from p.*

Proof. An immediate consequence of Lemma 4. □

Corollary 2 *Let p be a point on RC_i^l. If p is not visible from $lv(we_i)$, then the subchain of RC_i^l from p* clockwise *to $lv(we_i)$ defines a point p' on we_i such that the segment $\overline{lv(we_i)p'}$ is the maximal segment on we_i that is not visible from p.*

Proof. An immediate consequence of Lemma 3 and Corollary 1. □

Lemma 5 *Let p be a point on RC_i^r, and let we_j be the edge of WVE such that we_j does not contain p and that we_j is the first edge encountered among the edges of WVE when walking along RC_i^r from p* clockwise *to $lv(we_{i+1})$. Then if p is not visible from $lv(we_i)$, then a vertex of we_j must define a point p' on we_i such that the segment $\overline{lv(we_i)p'}$ is the maximal segment on we_i that is not visible from p (see Figure 8).*

Proof. Let p' be the point on we_i such that segment $\overline{lv(we_i)p'}$ is the maximal segment on we_i that is not visible from p. Note that p' can be $lv(we_i)$ because $lv(we_i)$ is not visible from p. By Lemma 3, the view between p and every point q on $\overline{lv(we_i)p'}$ can be blocked only by the chain along RC_i from p clockwise to q. If p' were defined by a point on the chain along RC_i from p clockwise to

$lv(we_j)$ excluding $lv(we_j)$, then p would have not been weakly visible from we_j, a contradiction. If p' were defined by a point on the chain along RC_i from $fv(we_j)$ clockwise to p' excluding $fv(we_j)$, then $lv(we_i)$ would have not been weakly visible from we_j, again a contradiction. Hence only the vertices of we_j can define p' on we_i for p. □

Note that in Corollary 2 and Lemma 5, point $p \in RC_i$ is visible from every point on the segment $\overline{fv(we_i)p'} \subset we_i$ (this follows from Corollary 1). For every point p on RC_i^r, by Lemma 5, point p' on we_i can be easily computed. For every point p on RC_i^l, by Corollary 2, point p' on we_i can be found out if the line segment that is on the internal convex path from $lv(we_i)$ to p along RC_i^l and that is adjacent to p is known.

We define a total order on the points of we_i, as follows: For every two points q' and q'' on we_i, $q' \leq q''$ iff segment $\overline{fv(we_i)q'}$ is contained by segment $\overline{fv(we_i)q''}$. Let edge we_i correspond to the interval $[fv(we_i), lv(we_i)]$. For every vertex v_k of RC_i, let $[lp_k, rp_k]$ be the interval on we_i such that segment $\overline{lp_k rp_k}$ is the maximal segment on we_i that is visible from v_k. We denote $[lp_k, rp_k]$ by I_k. Note that it is possible that $lp_k = rp_k$. For example, if the only point on we_i from which v_k is visible is $lv(we_i)$, then $I_k = [lv(we_i), lv(we_i)]$. The intervals I_k have the following property:

Lemma 6 *For every pair of consecutive vertices v_k and v_{k+1} of RC_i, $I_k \cap I_{k+1} \neq \emptyset$.*

Proof. There are three cases. If I_{k+1} is equal to $[lv(we_i), lv(we_i)]$, then I_k is also equal to $[lv(we_i), lv(we_i)]$, by Lemma 3 (otherwise, v_k would have not been weakly visible from we_i). If I_k is equal to $[lv(we_i), lv(we_i)]$ but I_{k+1} is not, then I_{k+1} must be equal to $[fv(we_i), lv(we_i)]$ (this also follows from Lemma 3). If both I_k and I_{k+1} are not equal to $[lv(we_i), lv(we_i)]$, then they must both contain $fv(we_i)$. □

From the intervals I_k of the vertices v_k on RC_i, we define a set of intervals on we_i, called the *characteristic intervals*, as follows: For every edge e_j on RC_i, let

$$CI_j = I_j \cap I_{j+1},$$

and call CI_j the *characteristic interval* of e_j. The next lemma illustrates the relation between $s(we_i)$ and the characteristic intervals for the edges of RC_i.

Lemma 7 *The shortest weakly visible subedge $s(we_i)$ on we_i must contain at least one point on interval CI_j, for every edge e_j on RC_i.*

Proof. This follows from the fact that edge e_j is completely visible from every point on interval CI_j (see Figure 9). □

The next section gives the sequential and parallel algorithms for computing the shortest weakly visible subedge s of P.

Figure 9: Edge e_j is completely visible from every point on CI_j.

4 Algorithms for the SWVS Problem

We are now ready to present the algorithms for computing the shortest weakly visible subedge s of P. The correctness of these algorithms is based on the observations made in Section 3. Conceptually, these algorithms are quite simple.

We need some simple notation (we only give that with respect to the chain RC_i). If vertex $lv(we_i)$ is nonconvex, then let r_i be the ray originating from $fv(we_i)$ and passing $lv(we_i)$, and let r_i first hit $bd(P) - we_i$ at point h_i. Denote the chain along $bd(P)$ from $lv(we_i)$ counterclockwise to h_i by RP_i (called the *right pocket* of we_i). The following properties of RP_i are easily seen to be true:

- The chain $RP_i - \{lv(we_i), h_i\}$ can intersect at most two edges of WVE (i.e., we_{i+1} and we_{i+2}), and if this is the case, then $C_i = \emptyset$.
- Point h_i is contained in chain RC_i, and h_i is the first point on RP_i that intersects the ray r_i, where r_i is viewed as a half-line (otherwise, some point on RP_i would have not been weakly visible from we_i).
- The only point on we_i from which every point on $RP_i - \{lv(we_i), h_i\}$ is visible is $lv(we_i)$.

For every vertex v_k of $RC_i - lv(we_i)$, let ICP_i^k be the internal convex path connecting $lv(we_i)$ and v_k, and let w_k be the line segment on ICP_i^k such that w_k is adjacent to v_k. Let r_i^k be the ray originating from v_k and containing w_k. Note that r_i^k must intersect we_i. Let the intersection point of r_i^k and we_i be ip_i^k.

The general procedure for solving the SWVS problem is as follows.

Algorithm SWVS.

Input. A simple polygon P of n vertices.

Output. The shortest weakly visible subedge s of P.

(1) Compute WVE for P. If $|WVE| < 7$, then compute $s(we_i)$ on every edge $we_i \in WVE$, find s from these $s(we_i)$'s, and stop.

(2) For every $we_i \in WVE$, perform the following computation on chain RC_i:

(2.1) Vertex $lv(we_i)$ is convex. For every vertex v_k on RC_i^l, compute the segment w_k on ICP_i^k (by Corollary 2) and the intersection point ip_i^k between r_i^k and we_i; let interval I_k be $[fv(we_i), ip_i^k]$. For every vertex v_k on RC_i^r, compute ip_i^k by Lemma 5, and let I_k be $[fv(we_i), ip_i^k]$.

(2.2) Vertex $lv(we_i)$ is nonconvex. For every vertex v_k on $RP_i - lv(we_i)$, let interval I_k be $[lv(we_i), lv(we_i)]$. For every vertex v_k on $RC_i^l - RP_i$ (resp., $RC_i^r - RP_i$), compute I_k as in Step (2.1), by using Corollary 2 (resp., Lemma 5).

(2.3) Compute the characteristic interval CI_j for every edge e_j on RC_i. Let the set of characteristic intervals so obtained be I_i^R.

(3) For every $we_i \in WVE$, perform computation similar to Step (2) on chain LC_i. Let the set of characteristic intervals so obtained be I_i^L.

(4) For every $we_i \in WVE$, compute $s(we_i)$ as follows: Let

$$\alpha_i = \max\{lp_k \mid [lp_k, rp_k] \in I_i^R \cup I_i^L\},$$

and

$$\beta_i = \min\{rp_k \mid [lp_k, rp_k] \in I_i^R \cup I_i^L\}.$$

If $\alpha_i \leq \beta_i$, then let $s(we_i)$ be *any point* on interval $[\alpha_i, \beta_i]$; otherwise, let $s(we_i) = [\beta_i, \alpha_i]$.

(5) Let $s = s(we_j)$, where

$$|s(we_j)| = \min\{|s(we_i)| \mid we_i \in WVE\}.$$

Lemma 8 **Algorithm SWVS** *can be implemented in $O(n)$ time sequentially, and in $O(\log n)$ time using $O(n/\log n)$ CREW PRAM processors in parallel.*

Proof. The sequential implementation of **Algorithm SWVS** is as follows. Step (1) is performed by first using [20, 21] and then using [1, 7], in $O(n)$ time. Steps (2) and (3) are implemented by using [1, 7]. That these steps take $O(n)$ time follows from the fact that each chain C_j involves in the computation for at most six edges $we_i \in WVE$. Steps (4) and (5) can be easily implemented in $O(n)$ time. Therefore, the overall time complexity is $O(n)$.

The parallel implementation does the following. Step (1) is done by first using [5] and then using [6], in $O(\log n)$ time using $O(n/\log n)$ CREW PRAM processors. Steps (2) and (3) are performed by using [5] and parallel prefix [16, 17]. Steps (4) and (5) are easily handled by using parallel prefix [16, 17]. Therefore, the parallel algorithm runs in $O(\log n)$ time using $O(n/\log n)$ CREW PRAM processors. □

5 The Rectilinear Case

In this section, we study the special case of the SWVS problem where the polygons are rectilinear (i.e., each edge of the polygons is either vertical or horizontal). We present very simple and optimal solutions to solve this case, both sequentially and in parallel. As for the general SWVS problem, we also give interesting geometric observations for solving the rectilinear case. These geometric observations enable us to design extremely simple algorithms for this case. The parallel model we use in this section is the EREW PRAM.

In the rest of this section, we let P be a rectilinear simple polygon of n vertices. For a subchain $C(i, i+3) = (e_i,\ e_{i+1},\ e_{i+2})$ of $bd(P)$, we call $C(i, i+3)$ a *concave chain* of P iff edge e_{i+1} is nonconvex (i.e., the interior angles of P at v_{i+1} and v_{i+2} are both greater than π), and call edge e_{i+1} the *center edge* of $C(i, i+3)$. Let the line containing an edge e_j be $l(e_j)$. We say that a concave chain $C(i, i+3)$ is *upward* (resp., *downward*, *leftward*, *rightward*) if e_{i+1} is horizontal (resp., horizontal, vertical, vertical) and if no point on $C(i, i+3)$ is strictly above (resp., below, to the left of, to the right of) line $l(e_{i+1})$.

For every vertex v_i of P, if v_{i+1} (resp., v_{i-1}) is nonconvex, then let r_i^+ (resp., r_i^-) be the ray starting at v_i and containing e_i (resp., e_{i-1}). If ray r_i^+ (resp., r_i^-) is associated with v_i, then let $h(r_i^+)$ (resp., $h(r_i^-)$) be the point on $bd(P) - e_i$ (resp., $bd(P) - e_{i-1}$) that is first hit by r_i^+ (resp., (r_i^-)).

A subchain C of $bd(P)$ is said to be *x-monotone* (resp., *y-monotone*) iff the intersection between C and every vertical (resp., horizontal) line is a single connected component. A subchain C' of $bd(P)$ is said to be a *staircase* iff C' is both x-monotone and y-monotone. Polygon P is said to be *x-monotone* (resp., *y-monotone*) iff $bd(P)$ can be partitioned into two x-monotone (resp., y-monotone) chains.

Let $C(i, i+3)$ be an upward concave chain (the other cases are similar). Then the following properties of $C(i, i+3)$ can be easily seen to hold (see Figure 10).

- The only possible weakly visible edge of P on $C(i, i+3)$ is the center edge e_{i+1} of $C(i, i+3)$.
- If $e_{i+1} \in WVE$, then the following are true:
  1. The subchain of $bd(P)$ from $h(r_{i+3}^-)$ counterclockwise to $h(r_i^+)$ is x-monotone.

Figure 10: Illustrating some properties of $C(i, i+3)$.

2. Both v_i and v_{i+3} are convex.

3. The subchain of $bd(P)$ from v_i (resp., v_{i+3}) clockwise (resp., counterclockwise) to $h(r^-_{i+2})$ (resp., $h(r^+_{i+1})$) is a staircase, and the subchain of $bd(P)$ from $h(r^-_{i+2})$ (resp., $h(r^+_{i+1})$) clockwise (resp., counterclockwise) to $h(r^+_i)$ (resp., $h(r^-_{i+3})$) is a staircase, as shown in Figure 10.

4. $s(e_{i+1}) = e_{i+1}$.

The following lemmas are useful for our algorithms.

Lemma 9 *If polygon P has two concave chains $C(i, i+3)$ and $C(j, j+3)$, where $C(i, i+3)$ is either upward or downward and $C(j, j+3)$ is either leftward or rightward, then P is not weakly visible from any of its edges.*

Proof. For any vertical edge e' of P, there must be some points on either e_i or e_{i+2} (both are vertical) that are not weakly visible from e'. For any horizontal edge e'' of P, there must be some points on either e_j or e_{j+2} (both are horizontal) that are not weakly visible from e''. Hence the lemma follows. □

Corollary 3 *If polygon P is neither x-monotone nor y-monotone, then P is not weakly visible from any of its edges.*

Proof. If P is not x-monotone, then it must have a leftward or rightward concave chain. If P is not y-monotone, then it must have an upward or downward concave chain. That P is not weakly visible follows immediately from Lemma 9. □

Lemma 10 *Suppose that polygon P has only upward and downward (resp., leftward and rightward) concave chains. Let $C(i, i+3)$ be such a concave chain. Then WVE consists of at most* two *edges of P: (i) the center edge e_{i+1} of $C(i, i+3)$, and (ii) the edge e_j such that e_j contains* both *the points $h(r^+_i)$ and $h(r^-_{i+3})$ (if such an edge e_j exists).*

Proof. Observe that the segments $\overline{v_{i+1}h(r_i^+)}$ and $\overline{v_{i+2}h(r_{i+3}^-)}$ together partition P into three subpolygons. Because of edges e_i and e_{i+2}, only those edges of P that intersect all these three subpolygons can possibly be weakly visible edges of P. The only edges of P that intersect the three subpolygons are e_{i+1} and e_j (if such an e_j exists). □

Lemma 11 *Suppose that polygon P is x-monotone and has two distinct upward (resp., downward) concave chains $C(i, i+3)$ and $C(j, j+3)$. Then P can possibly be weakly visible from at most* one *edge e_k, such that e_k contains all the points $h(r_i^+)$, $h(r_{i+3}^-)$, $h(r_j^+)$, and $h(r_{j+3}^-)$, if such an edge e_k exists.*

Proof. The segments $\overline{v_{i+1}h(r_i^+)}$, $\overline{v_{i+2}h(r_{i+3}^-)}$, $\overline{v_{j+1}h(r_j^+)}$, and $\overline{v_{j+2}h(r_{j+3}^-)}$ together partition P into five subpolygons. Any weakly visible edge of P must intersect all these five subpolygons, and edge e_k is the only such candidate (if it exists). □

Lemma 12 *Let C be a staircase chain on $bd(P)$. Then if C has more than* four *edges, then no edge on C can be in WVE.*

Proof. Such a staircase C must have at least two nonconvex vertices u and v. Let e be an arbitrary edge of C. then e can be adjacent to at most one of u and v (say, v). For e, there must be some point p on the edges of P adjacent to u such that p is not weakly visible from e. Hence $e \notin WVE$. □

Let e_j be an edge in *WVE* such that e_j is on a staircase of $bd(P)$ and that e_j is not the center edge of any concave chain of P. There are two possible cases for e_j: Either both vertices of e_j are convex or exactly one vertex of e_j is convex. We consider first the case where exactly one vertex of e_j is convex. WLOG, let v_j be convex and v_{j+1} be nonconvex (the case where v_j is nonconvex and v_{j+1} is convex is symmetric). It is easy to see that the following properties of e_j hold:

- Vertices v_{j-1} and v_{j+2} are both be convex. Therefore, e_j must be adjacent to an ending edge of a maximal staircase of $bd(P)$.
- Suppose that line $l(e_j)$ is horizontal (the other case is similar). Then the subchain of $bd(P)$ from $h(r_{j+2}^-)$ counterclockwise to v_{j-1} is x-monotone.
- The subchain of $bd(P)$ from v_{j+2} counterclockwise to $h(r_j^+)$ is a staircase, and the subchain of $bd(P)$ from $h(r_j^+)$ counterclockwise to $h(r_{j+2}^-)$ is a staircase.
- Let $H_j = \{h(r_k^-) \mid v_k$ is on the subchain of $bd(P)$ from $h(r_{j+2}^-)$ counterclockwise to v_{j-1} and v_{k-1} is nonconvex$\}$. If $H_j = \emptyset$, then $s(e_j) = v_{j+1}$. Otherwise, let α_j be the point in H_j that is closest to v_j among all the points in H_j; then $s(e_j) = \overline{\alpha_j v_{j+1}}$.

The case where both vertices of $e_j \in WVE$ are convex has the following properties:

- Vertices v_{j-1} and v_{j+2} are both convex. Hence e_j is an ending edge of a maximal staircase on $bd(P)$.
- The subchain of $bd(P)$ from v_{j+2} counterclockwise to v_{j-1} (i.e., $C(j+2, j-1)$) is monotone with respect to line $l(e_j)$.
- Let $RH_j = \{h(r_k^-) \mid v_k$ is on $C(j+2, j-1)$ and v_{k-1} is nonconvex$\}$, and $LH_j = \{h(r_k^+) \mid v_k$ is on $C(j+2, j-1)$ and v_{k+1} is nonconvex$\}$. Let α_j (resp., β_j) be the point in RH_j (resp., LH_j) that is closest to v_j (resp., v_{j+1}) among all the points in RH_j (resp., LH_j). If both α_j and β_j do not exist, then $s(e_j)$ can be any point on e_j. If exactly β_j (resp., α_j) does not exist, then $s(e_j)$ can be any point on the segment $\overline{\alpha_j v_j}$ (resp., $\overline{\beta_j v_{j+1}}$). If both α_j and β_j exist, then $s(e_j) = \overline{\alpha_j \beta_j}$.

Lemma 13 *If $WVE \neq \emptyset$, then $|WVE| = O(1)$.*

Proof. There are two cases. If P is x-monotone or y-monotone but not both, then by Lemmas 10 and 11, $|WVE|$ can be at most 2. If P is both x-monotone and y-monotone, then $bd(P)$ has at most *four* maximal subchains such that each subchain is a staircase. There are totally 4 ending edges on these four maximal staircases and there are at most 8 edges that are adjacent to the ending edges of these four staircases. Hence in this case, the lemma follows from the properties of the edges in WVE that are on a staircase of $bd(P)$. □

Our results on solving the rectilinear case of the SWVS problem are summarized in the following lemma.

Lemma 14 *Given a rectilinear polygon P, there are extremely simple and optimal algorithms for computing, both sequentially and in parallel, (i) WVE, and (ii) the shortest weakly visible subedge s of P. The sequential algorithm runs in $O(n)$ time, and the parallel algorithm runs in $O(\log n)$ time using $O(n/\log n)$ EREW PRAM processors.*

Proof. The sequential algorithm easily follows from the above observations. It only needs to do the following: (1) Check the monotonicity of P with respect to the x and y axes, (2) identify the $O(1)$ edges of WVE, and (3) find $s(e)$ on each edge $e \in WVE$. The parallel algorithm is also very straightforward and makes use of only simple EREW PRAM operations such as parallel prefix [16, 17]. The details of these algorithms are left to the reader as an exercise. □

6 Conclusion

We continue the study of the weak visibility problems on simple polygons that were first considered by Avis and Toussaint [1] and then by many others [5, 6, 7, 10, 20, 21]. We present new geometric observations on the weak visibility of simple polygons. We show that, by using these geometric observations and the previously known algorithms in [1, 5, 6, 7, 20, 21], the problem of computing the shortest weakly visible subedge of a simple polygon can be solved optimally, both sequentially and in parallel. Our sequential algorithm for this problem runs in $O(n)$ time, and our parallel algorithm runs in $O(\log n)$ time using $O(n/\log n)$ CREW PRAM processors. We also give geometric observations that lead to extremely simple and optimal solutions to the case of this problem where the polygons are rectilinear. We expect the observations that we present to be useful in solving other visibility problems.

References

[1] D. Avis and G. T. Toussaint. "An optimal algorithm for determining the visibility polygon from an edge," *IEEE Trans. Comput.*, C-30 (12) (1981), pp. 910–914.

[2] B. K. Bhattacharya, D. G. Kirkpatrick, and G. T. Toussaint. "Determining sector visibility of a polygon," *Proc. 5-th Annual ACM Symp. Computational Geometry*, 1989, pp. 247–254.

[3] B. K. Bhattacharya, A. Mukhopadhyay, and G. T. Toussaint. "A linear time algorithm for computing the shortest line segment from which a polygon is weakly externally visible," *Proc. Workshop on Algorithms and Data Structures (WADS'91)*, 1991, Ottawa, Canada, pp. 412–424.

[4] B. Chazelle and L. J. Guibas. "Visibility and intersection problems in plane geometry," *Discrete and Computational Geometry*, 4 (1989), pp. 551–581.

[5] D. Z. Chen. "An optimal parallel algorithm for detecting weak visibility of a simple polygon," *Proc. of the Eighth Annual ACM Symp. on Computational Geometry*, 1992, pp. 63–72.

[6] D. Z. Chen. "Parallel techniques for paths, visibility, and related problems," Ph.D. thesis, Technical Report No. 92-051, Dept. of Computer Sciences, Purdue University, July 1992.

[7] Y. T. Ching, M. T. Ko, and H. Y. Tu. "On the cruising guard problems," Technical Report, 1989, Institute of Information Science, Academia Sinica, Taipei, Taiwan.

[8] J. I. Doh and K. Y. Chwa. "An algorithm for determining visibility of a simple polygon from an internal line segment," *Journal of Algorithms*, 14 (1993), pp. 139–168.

[9] H. ElGindy. "Hierarchical decomposition of polygon with applications," Ph.D. thesis, McGill University, 1985.

[10] S. K. Ghosh, A. Maheshwari, S. P. Pal, S. Saluja, and C. E. V. Madhavan. "Characterizing weak visibility polygons and related problems," Technical Report No. IISc-CSA-90-1, 1990, Dept. Computer Science and Automation, Indian Institute of Science.

[11] M. T. Goodrich, S. B. Shauck, and S. Guha. "Parallel methods for visibility and shortest path problems in simple polygons (Preliminary version)," *Proc. 6-th Annual ACM Symp. Computational Geometry*, 1990, pp. 73–82.

[12] L. J. Guibas, J. Hershberger, D. Leven, M. Sharir, and R. E. Tarjan. "Linear time algorithms for visibility and shortest paths problems inside triangulated simple polygons," *Algorithmica*, 2 (1987), pp. 209–233.

[13] P. J. Heffernan and J. S. B. Mitchell. "Structured visibility profiles with applications to problems in simple polygons," *Proc. 6-th Annual ACM Symp. Computational Geometry*, 1990, pp. 53–62.

[14] J. Hershberger. "Optimal parallel algorithms for triangulated simple polygons," *Proc. 8-th Annual ACM Symp. Computational Geometry*, 1992, pp. 33–42.

[15] Y. Ke. "Detecting the weak visibility of a simple polygon and related problems," manuscript, Dept. of Computer Science, The Johns Hopkins University, 1988.

[16] C. P. Kruskal, L. Rudolph, and M. Snir. "The power of parallel prefix," *IEEE Trans. Comput.*, C-34 (1985), pp. 965–968.

[17] R. E. Ladner and M. J. Fischer. "Parallel prefix computation," *Journal of the ACM*, 27 (4) (1980), pp. 831–838.

[18] D. T. Lee and A. K. Lin. "Computing the visibility polygon from an edge," *Computer Vision, Graphics, and Image Processing*, 34 (1986), pp. 1–19.

[19] S. H. Lee and K. Y. Chwa. "Some chain visibility problems in a simple polygon," *Algorithmica*, 5 (1990), pp. 485–507.

[20] J.-R. Sack and S. Suri. "An optimal algorithm for detecting weak visibility of a polygon," *IEEE Trans. Comput.*, C-39 (10) (1990), pp. 1213–1219.

[21] S. Y. Shin. "Visibility in the plane and its related problems," Ph.D. thesis, University of Michigan, 1986.

[22] G. T. Toussaint. "A linear-time algorithm for solving the strong hidden-line problem in a simple polygon," *Pattern Recognition letters*, 4 (1986), pp. 449–451.

[23] G. T. Toussaint and D. Avis. "On a convex hull algorithm for polygons and its applications to triangulation problems," *Pattern Recognition*, 15 (1) (1982), pp. 23–29.

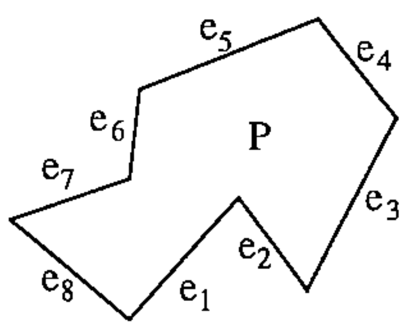
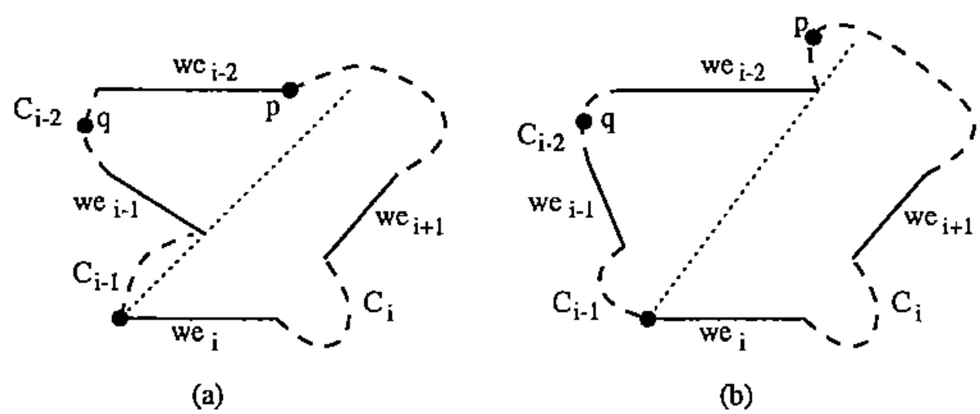


fig.1



we i-2
Ci-2
q
p
we i-1
we i+1
Ci-1
Ci
we i
(a)
p
we i-2
q
Ci-2
we i-1
we i+1
Ci-1
Ci
we i
(b)

fig.2

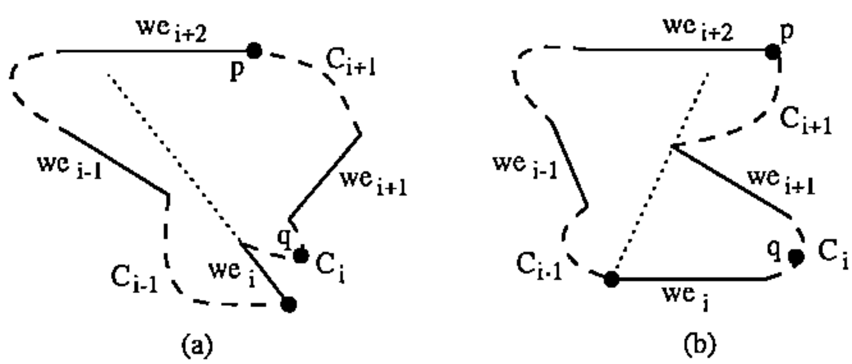

we i+2
p
C i+1
we i-1
we i+1
q
C i
C i-1
we i
(a)
we i+2
p
C i+1
we i-1
we i+1
q
C i
C i-1
we i
(b)

fig.3

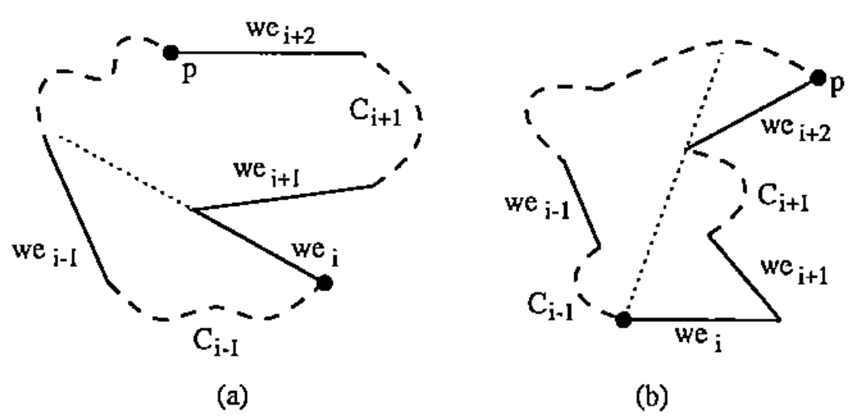


fig.4

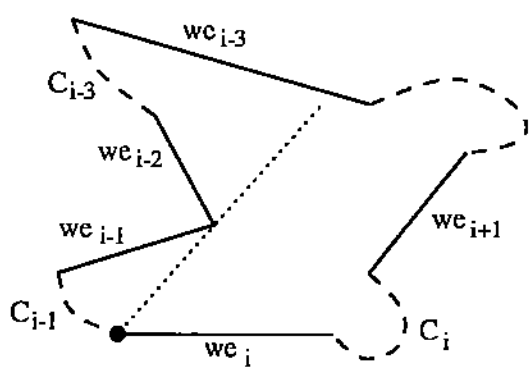


fig.5

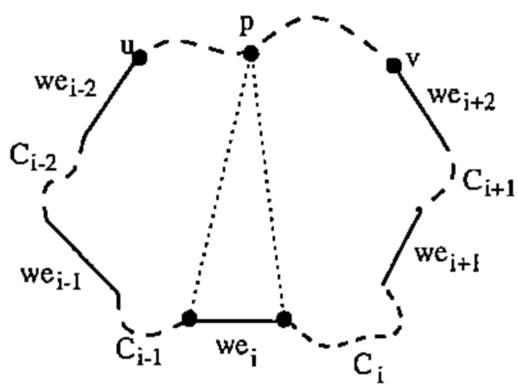


fig.6

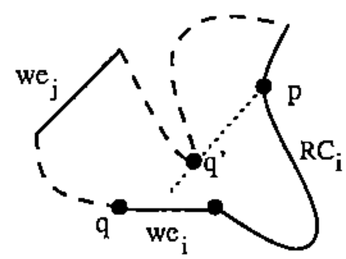


fig.7

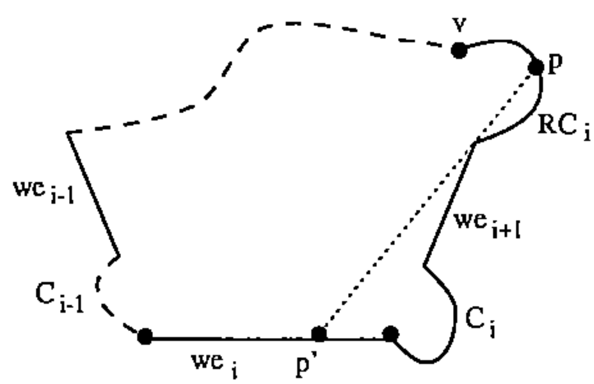


fig.8

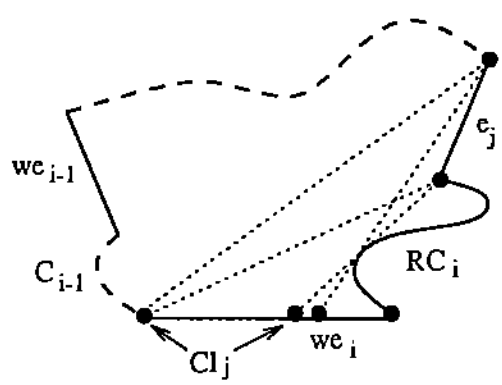


fig.9

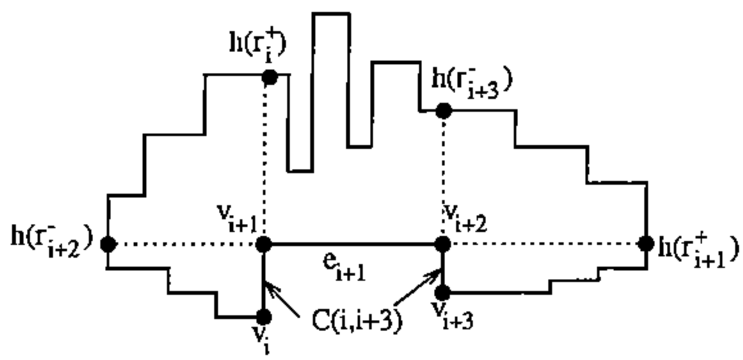


fig.10