Isomorphism of Spiral Polygons*

G. MacDonald and T. Shermer

School of Computing Science, Simon Fraser University,
Burnaby, British Columbia, Canada V5A 1S6

Abstract. We call two polygons *isomorphic* if there is a one-to-one mapping between their *points* (not vertices) that preserves visibility. In this paper we establish necessary and sufficient conditions for two spiral polygons to be isomorphic, and give an $O(n^2)$ algorithm to detect such isomorphism. We also show that the continuous graph of visibility on the points of a spiral polygon is an (uncountably infinite) interval graph, and that no other polygons have this property.

1. Introduction

Let P be a simple polygon (with interior). A vertex of P is called *reflex* if its interior angle is greater than 180° , and *convex* otherwise. Two points x and y in P are said to be visible if the line segment $\overline{xy}$ is contained in P . The *point visibility graph* of P , denoted $PVG(P)$, is the graph of the visibility relation [13]; i.e., $PVG(P) = (V, E)$ where

$$V = \{x \mid x \in P\} \quad \text{and} \quad E = \{(x, y) \mid \overline{xy} \subset P\}.$$

Point visibility graphs are *continuous graphs*: for any polygon, the point visibility graph has $\aleph_1$ vertices, and each vertex is an endpoint of $\aleph_1$ edges (for notational convenience, we use $\aleph_1$ to denote the cardinality of the reals; we neither need nor assume the continuum hypothesis). When the context is clear, a point in P and the corresponding node in $V(PVG(P))$ is referred to by the same name. However, when the context is ambiguous, the term *node* is used to refer to members of $V(PVG(P))$, while the terms *point* and *vertex* are reserved for referring to parts of the polygon. Similarly, the term *arc* is used for the members of $E(PVG(P))$, and *edge* is reserved for polygon or decomposition edges.

* This research was supported by the Natural Sciences and Engineering Research Council of Canada under Research Grant Number OGP0046218 and a Post-Graduate Scholarship.

Many geometric properties of polygons are graph-theoretic properties of their point visibility graphs; we call a property of this sort a *pure visibility property*. For example, link-diameter, link-center, convex cover, minimum guard set, and maximum hidden set are pure visibility properties because in a polygon P they correspond to (respectively) the diameter, center, clique cover, minimum dominating set, and maximum independent set of $PVG(P)$. Although most authors have not been thinking in terms of point visibility graphs or pure visibility properties, there have been many recent investigations into the computational aspects of individual pure visibility properties [13], [4], [10], [7], [8], [15], [12].

Two point visibility graphs G and G' are said to be *isomorphic* if there is a one-to-one mapping between $V(G)$ and $V(G')$ that preserves adjacency. For convenience, we also call two polygons P and P' isomorphic if the point visibility graphs $PVG(P)$ and $PVG(P')$ are isomorphic, i.e., there is a one-to-one mapping between the points in P and the points in P' that preserves the visibility relation. For example, all convex polygons are isomorphic, and all polygons with exactly one reflex vertex are isomorphic.

The question of whether two polygons are isomorphic is of interest because all pure visibility properties are invariant over classes of isomorphic polygons. An isomorphism algorithm for simple polygons would be the geometric equivalent of an isomorphism algorithm for all graphs. In this paper we begin to address the polygon isomorphism problem, presenting an $O(n^2)$ algorithm for determining isomorphism for *spiral* polygons. This algorithm proceeds by computing a natural polynomial-sized decomposition of the polygon (called the *cell decomposition*), which, as we show, uniquely determines the point visibility graph.

The visibility polygon of a point $x \in P$, denoted $VP(x)$, is the set of all points of P that are visible to x , i.e., $VP(x) = \{y \mid \overline{xy} \subset P\}$ (note that in some cases the visibility polygon fails to be a polygon, e.g., $VP(v)$ in Fig. 7). The neighborhood $N(x)$ of a node $x \in V(PVG(P))$ consists of the nodes whose corresponding points comprise $VP(x)$. The *kernel* of P , denoted $K(P)$, is that set of points that have all of P as their visibility polygon, i.e., $K(P) = \{x \in P \mid VP(x) = P\}$. Kernel points of P correspond to the *universal vertices* of $PVG(P)$. The kernel of a polygon is usually empty, and when it is not, we say that the polygon is *star shaped*.

One property of the kernel is that if $x, y \in K(P)$, then $VP(x) = VP(y)$. This same property may hold true of other subsets of P ; we call such subsets *semikernels*. Formally, a *semikernel* in P is a maximal set of points that all have the same visibility polygon. Thus, if the kernel of a polygon is nonempty, it is a semikernel. We call a semikernel *trivial* if it consists of exactly one point, and *nontrivial* if it consists of an uncountably infinite number of points (note that these are the only two cases). Two points in the same semikernel are indistinguishable in the point visibility graph, and thus nontrivial semikernels are treated specially in our analysis.

A polygon P is called *spiral* if its vertices, in the order they appear along the boundary, can be expressed as a sequence $u_1, u_2, \dots, u_r, v_1, v_2, \dots, v_{n-r}$ where the u_i 's are reflex and the v_i 's are convex. Note that n is therefore the number of vertices in P , and r is the number of reflex vertices. For convenience, we assume that the vertices are given in counterclockwise order, and let $u_0 = v_{n-r}$ and $u_{r+1} = v_1$. The *reflex chain* of P is the polygonal chain $u_0, u_1, \dots, u_{r+1}$ and the *convex chain* is the polygonal chain $v_1, v_2, \dots, v_{n-r}$.

Fig. 1. The left and right end regions.

The line segment $\overline{u_0 u_1}$, extended from u_1 until it intersects the convex chain at a point q , divides P into two subpolygons. One of these subpolygons contains u_0 , and this one is called the *left end region* of P . The *right end region* is defined similarly. Figure 1 illustrates these definitions.

The *forward extension* from a point x in a spiral polygon P is the line segment from x through the highest-numbered reflex vertex u_i seen by x , extended until it intersects the convex chain at x' (see Fig. 2). This definition is used unless x is in (the relative interior of) the right end region, in which case x has no forward extension. The segment $\overline{u_i x'}$ is called the *forward window* of x , and the point x' is called the *forward extension point* of x . Similarly, we define the *backward extension* from x (through the lowest-numbered reflex vertex seen), the *backward window* of x , and the *backward extension point* of x . If x is in the left end region, then x has no backward extension.

The *first-order cell decomposition* of P , denoted $CD_1(P)$, is the diagram consisting of the boundary of P together with the forward and backward extensions of the vertices of the reflex chain; a first-order cell decomposition is shown in Fig. 3. Let B be the set of extension points at the ends of the extensions in $CD_1(P)$. The *second-order cell decomposition* of P , denoted $CD_2(P)$, is obtained by adding the forward and backward extension of each member of B to $CD_1(P)$. Note that, for each member of B , one of the extensions will already be in $CD_1(P)$; this extension is not added again. (In fact, if the member of B is a forward extension point, only its forward extension need be produced, and if it is a backward extension point, only its backward extension need be produced.) A second-order cell decomposition is shown in Fig. 4. The third- and higher-order cell decompositions are determined in a similar manner, by adding the forward and

Fig. 2. The extensions of a point.

Fig. 3. A first-order cell decomposition.

backward extensions of each extension point in the the previous cell decomposition. A cell decomposition is well defined for each possible order. However, it is possible that, for a given order j , $CD_j(P) = CD_k(P)$ for all $k > j$. Such a cell decomposition is called a *full cell decomposition* (or *the cell decomposition of P*) and is denoted $CD(P)$. A full cell decomposition is shown in Fig. 5.

We now show that it is always the case that $CD_r(P) = CD(P)$. Consider a vertex u_i on the reflex chain in P and an extension point p of u_i in $CD_1(P)$. By symmetry, assume without loss of generality that p is the forward extension point of u_i . Unless p is in the right end region, its forward extension $\overline{pp'}$ will be in $CD_2(P)$. Similarly, unless p' is in the right end region, its forward extension $\overline{p'p''}$ will be in $CD_3(P)$, etc. Since each forward extension goes through the highest-numbered reflex vertex seen, and the extension point at the end of a forward extension (unless in the right end region) sees a reflex vertex numbered higher than the reflex vertex that the extension passes through, the reflex vertices that these successive forward extensions from u_i pass through are encountered in strictly increasing order. Thus, we can successively extend u_i at most r times before the extension point ends up in the right end region; this means that in $CD_r(P)$, there are no new extensions left to make, so $CD_r(P)$ is the full cell decomposition (of course, it is also possible that some $CD_s(P)$ for $s < r$ is also the full cell decomposition).

Two cell decompositions $CD(P)$ and $CD(P')$ are called *isomorphic* if the elements (cells, edges, and vertices) of the two cell decompositions can be placed in one-to-one

Fig. 4. A second-order cell decomposition.

Fig. 5. A full cell decomposition.

correspondence such that adjacency is preserved. In this definition we are not concerned with the convex vertices of the polygon; two consecutive places where extensions meet the convex chain are considered to be connected by a single edge of the cell decomposition, regardless of how many (or how few) convex vertices intervene. Figure 6 shows two isomorphic cell decompositions.

We establish that two spiral polygons are isomorphic iff their cell decompositions are isomorphic, and that cell decompositions have a concise representation. The remainder of this paper is organized as follows: in Section 2 we show that the point visibility graph of a spiral polygon completely determines the cell decomposition. In Section 3 we show the converse: the cell decomposition of a spiral completely determines its point visibility graph. Section 4 contains the details of the algorithm for spiral polygon isomorphism, and Section 5 investigates the characterization of spiral point visibility graphs, showing that a point visibility graph is from a spiral polygon iff it is an interval graph. Concluding remarks are made in Section 6.

2. The Point Visibility Graph Determines the Cell Decomposition

In this section, we show that the cell decomposition of a spiral polygon can be obtained from the point visibility graph. We first characterize the semikernels of a spiral polygon; this characterization is used to identify the points of the end regions and the reflex chain

Fig. 6. Isomorphic cell decompositions.

in the point visibility graph. For conciseness, in the following lemma we adopt the convention that the left end region includes neither endpoint of the forward window of u_0 , and the right end region includes neither endpoint of the backward window of u_r .

Lemma 2.1. *Let P be a spiral polygon. Then each nontrivial semikernel of P is either:*

- *the kernel of P , if one exists and is not a single point,*
- *an edge of the reflex chain, open at the reflex vertices,*
- *the intersection of the left end region with a line segment from a point p on the convex boundary of the left end region to the highest-numbered reflex vertex that p sees, or*
- *the intersection of the right end region with a line segment from a point p on the convex boundary of the right end region to the lowest-numbered reflex vertex that p sees.*

Proof. The visibility polygon of a point x in P is formed by partitioning P with the forward and backward windows of x , if such windows exist. Thus, for two points x and y to have the same visibility polygon (i.e., to be in the same semikernel), they must have the same forward and backward windows.

If x and y both have neither forward extensions nor backward extensions, then both x and y are in the kernel, which is nontrivial, and hence are in the same nontrivial semikernel. If x and y both have forward extensions but not backward ones, then they both lie in the left end region of P , on the same line segment from the highest-numbered reflex vertex seen. If x and y both have backward extensions but not forward ones, then they both lie in the right end region of P , on the same line segment from the lowest-numbered reflex vertex seen. If x and y both have both backward and forward extensions, then they both lie on the same line segment from the lowest-numbered reflex vertex seen, and on the same line segment from the highest-numbered reflex vertex seen. This is only possible on a segment between two adjacent reflex vertices. If x has a backward extension and y does not, then y sees u_0 and x does not, and therefore x and y do not have the same visibility polygon. A similar argument applies whenever x and y do not both have extensions in the same directions. $\square$

We now state the main result of this section.

Theorem 2.2. *Let G be the point visibility graph of some spiral polygon P . Then the cell decomposition of P can be determined by examining G .*

The remainder of this section contains the proof of this theorem. The proof is constructive and consists of four main parts:

1. Determining the points in the reflex chain and finding a relative ordering of the edges along the chain.
2. Determining a forward and a backward ordering on the nodes of G .

3. Determining forward and backward extensions of nodes of G .
4. Determining the cell decomposition.

2.1. Extracting the Reflex Chain

A node x of a graph is called *simplicial* if $N(x)$ is a clique. The simplicial vertices of a point visibility graph correspond to the points of the polygon that have a convex visibility polygon (which we call *simplicial points*).

The following lemma helps us define where simplicial points may lie in a spiral. It is stated in terms of *clear visibility* [1]; two points x, y in a polygon are said to be clearly visible if the open segment $\overline{xy}$ is entirely in the *interior* of P .

Lemma 2.3. *In any polygon, if a point x is simplicial, then no reflex vertex is clearly visible from x .*

Proof. Suppose that x is simplicial and that x and some reflex vertex r are clearly visible. Consider a small region that is the intersection of $VP(x)$ and a small neighborhood of r . In this region there are two points (one on either “side” of r) that are not visible. This is a contradiction. $\square$

The converse may fail to be true in polygons having degeneracies, such as a reflex vertex that lies on the extension of an edge through a reflex endpoint (see Fig. 7).

In a spiral polygon the only points from which a reflex vertex is not clearly visible are the points along the reflex chain, excluding the actual reflex vertices (which are clearly visible to themselves). As all of these points are in fact simplicial, we obtain all of the points on the reflex chain (except the reflex vertices) by collecting the simplicial nodes of the given point visibility graph G .

Let S_V be the set of simplicial nodes of G . If $S_V = V(G)$, then P is a convex polygon and its cell decomposition contains only a single region, all of P . Thus, we are done. We henceforth assume that P is nonconvex.

Partition the set S_V into a minimum number of classes, with the members of each class all having identical neighborhoods. Then, by definition, each class is a semikernel of P . As S_V consists of the points on the edges of the reflex chain (except the reflex vertices), by Lemma 2.1 we know that we obtain one class for each edge in the reflex

Fig. 7. The converse of Lemma 2.3 does not hold.

Fig. 8. A region R_i .

chain. Thus, we have now determined the number of reflex vertices—if we obtained $r + 1$ classes, then P contains r reflex vertices. Call these classes $E_0, E_1, \dots, E_r$.

We now must find the reflex vertices, and order the classes $E_0, E_1, \dots, E_r$ as they appear along the boundary of P . Let R_i be $N(E_i)$; i.e., $R_i = N(v)$ for any $v \in E_i$ (see Fig. 8). Let Q be the set of nodes of G whose neighborhoods are exactly the union of two of the R_i 's; i.e., $Q = \{v \in V(G) \mid N(v) = R_i \cup R_j \text{ for some } i \neq j\}$.

Lemma 2.4. *For all $q \in Q$, $N(q) = R_i \cup R_j$ where E_i and E_j are sets corresponding to reflex edges that are adjacent in P .*

Proof. Suppose that there is some $q \in Q$ such that $N(q) = R_i \cup R_j$ where E_i and E_j are sets corresponding to reflex edges that are not adjacent in P . To see the points of E_i , which are in R_i , a point must be in R_i . Therefore, $q \in R_i \cap R_j$. As E_i and E_j do not correspond to adjacent reflex edges there is some E_k that corresponds to a reflex edge that is between the edges corresponding to E_i and E_j on the reflex chain. Any point in $R_i \cap R_j$ will see the points of E_k (see Fig. 9). However, E_k is in neither of R_i and R_j , and thus $N(q)$ contains points other than $R_i \cup R_j$, a contradiction. $\square$

Since there are only r pairs of adjacent reflex edges in P , this lemma implies that we can partition Q into at most r semikernels $Q_1, Q_2, \dots, Q_t$ (some of these semikernels may be trivial).

Fig. 9. q sees E_k .

Fig. 10. The regions in a spiral polygon having one reflex vertex.

Observe now that the neighborhood of a reflex vertex in a spiral is exactly the union of the neighborhoods of its adjacent edges: each reflex vertex is in Q . As each reflex vertex has a different visibility polygon, each reflex vertex is in a distinct Q_i . Therefore there must be exactly r semikernels $Q_1, Q_2, \dots, Q_r$, one corresponding to each reflex vertex.

If $r = 1$, then the semikernel $Q_1 = Q$ is nontrivial and corresponds to the kernel of P . The cell decomposition then is composed of three regions: Q , $R_1 \setminus Q$, and $R_2 \setminus Q$ (see Fig. 10). We can report this and are therefore done in the case that $r = 1$.

In the general case ($r \geq 2$), we can apply Lemma 2.1 to see that there will be two nontrivial semikernels (corresponding to u_1 and u_r) and $r - 2$ trivial semikernels amongst the Q_i 's.

Two of the R_i 's, say R_j and R_k , are each only used once to define a Q_i , all other R_i 's are used exactly twice. The two Q_i defined by R_j and R_k are the nontrivial semikernels and correspond to u_1 and u_r . Similarly, E_j and E_k correspond to the edges of P at the ends of the reflex chain. Without loss of generality, assume that E_j corresponds to $\overline{u_0 u_1}$.

We can now easily obtain the ordering of the E_i 's and the Q_i 's along the boundary of P . We start with E_j (the edge $\overline{u_0 u_1}$), and stop when we have determined the corresponding reflex vertices to all of the Q_i 's and the corresponding reflex edges to all of the E_i 's. In the general step, we have already determined some E_i 's and Q_i 's, and we currently have an E_t corresponding to an edge $\overline{u_s u_{s+1}}$. The vertex u_{s+1} of P is in one of the (one or) two Q_i 's defined by E_t . If E_t helps to define only one Q_i , then $E_t = E_j$ and this Q_i corresponds to u_1 . If E_t helps to define two Q_i 's, then one of these will have already been placed in the ordering. The other, not yet ordered, Q_i corresponds to u_{s+1} . Furthermore, the neighborhood of this Q_i is the union of the neighborhood of E_t and some other $E_{t'}$. This $E_{t'}$ corresponds to the edge $\overline{u_{s+1} u_{s+2}}$, and we repeat the general step with $E_{t'}$ under consideration (unless $s + 1 = r$, in which case we are done).

We can now assume that the Q_i 's, E_i 's, and R_i 's have been re-indexed to reflect this ordering (i.e., Q_1 corresponds to u_1 , E_0 corresponds to $\overline{u_0 u_1}$, etc.).

So far, we have found all of the points in the reflex chain, and we have an ordering of the edges and reflex vertices along the boundary of P . We do not know if the edges are ordered clockwise or counterclockwise along P . However, by assuming that E_j helped determine u_1 , we have assumed a counterclockwise ordering. The alternative, where E_j determines u_r , leads to a construction that is symmetric under reflection to the one

that we obtain. As cell decompositions that are reflections of each other are considered isomorphic under our definition, we do not need to treat the two cases separately.

2.2. Determining Forward and Backward Orderings

We now define two orderings on the nodes of G . These orderings correspond to how much a point sees to either “side” of the polygon—a node will be greater than another if it sees more.

First we define the forward ordering. We use the notations $<_f$, $\leq_f$, $=_f$, $\geq_f$, $>_f$ to denote the usual relations in this ordering.

In the forward ordering the nodes in R_r are all equal and will be the maximum elements. Consider the nodes in $R_{r-1} \setminus R_r$; these nodes are all less than the nodes in R_r . To attain an ordering relation between two nodes x, y in $R_{r-1} \setminus R_r$, we examine the sets $N(x) \cap R_r$ and $N(y) \cap R_r$. If $N(x) \cap R_r \subset N(y) \cap R_r$, then $x <_f y$. If $N(x) \cap R_r = N(y) \cap R_r$, then $x =_f y$. Finally, if $N(x) \cap R_r \supset N(y) \cap R_r$, then $x >_f y$.

Note that these are the only three possibilities; it cannot be that $N(x)$ contains a node of $R_r \setminus N(y)$ and that $N(y)$ contains a node of $R_r \setminus N(x)$. The set $N(x) \cap R_r$ can be constructed by partitioning R_r with the forward window of x , and $N(y) \cap R_r$ can be constructed by partitioning R_r with the forward window of y (see Fig. 11). If $N(x)$ contains a node of $R_r \setminus N(y)$ and $N(y)$ contains a node of $R_r \setminus N(x)$, then these partitioning segments (forward windows) must cross in R_r . However, both of these partitioning segments start at the vertex u_r , and therefore cannot cross in R_r .

Next, we consider $R_{r-2} \setminus R_{r-1}$. All of the nodes in this set are less than all of the nodes in R_{r-1} (and thus less than all of the nodes in $R_{r-1} \cup R_r$). Two nodes x, y in $R_{r-2} \setminus R_{r-1}$ are related as follows: if $N(x) \cap R_{r-1} \subset N(y) \cap R_{r-1}$, then $x <_f y$; if $N(x) \cap R_{r-1} = N(y) \cap R_{r-1}$, then $x =_f y$; and if $N(x) \cap R_{r-1} \supset N(y) \cap R_{r-1}$, then $x >_f y$. Again these are the only possibilities, for the same reason as before.

We proceed in a similar fashion for the rest of the nodes, considering each $R_i \setminus R_{i+1}$ in turn, in order of decreasing index. All of the nodes in this set are less than all of the nodes in R_{i+1} , and are related amongst themselves by comparing neighborhoods intersected with R_{i+1} . We have thus obtained an ordering on all of the nodes of G .

Fig. 11. Obtaining $N(x) \cap R_{i+1}$ and $N(y) \cap R_{i+1}$.

We may obtain the backward ordering in a similar fashion. We start with the nodes of R_0 being all equal and maximal. We consider each other $R_i \setminus R_{i-1}$ in order of increasing index. All nodes in $R_i \setminus R_{i-1}$ are less than all of the nodes in R_{i-1} , and are related amongst themselves by comparing neighborhoods intersected with R_{i-1} . We use the notations $<_b$, $\leq_b$, $=_b$, $\geq_b$, $>_b$ for this relation.

2.3. Finding Extensions

We now show how to find the forward and backward extensions of nodes of G . Since the cell decomposition construction extends only the vertices of the reflex chain and extension points, we consider only these two cases. Furthermore, we only detail how to find forward extensions; finding backward extensions is symmetric.

First we show how to find the forward extensions of the vertices of the reflex chain. Suppose that we have some reflex vertex u_i (actually, what we have is an element of Q_i), and we wish to find $F(u_i)$, the nodes of G that correspond to points in the forward extension of u_i . Note that the points of $F(u_i)$ are all in R_i . If u_i is not u_0 , then $F(u_i)$ is exactly the set of nodes of R_i that are least in the backward ordering. If u_i is u_0 , then $F(u_i)$ consists of E_0 and one point from each semikernel in $R_0 \cap R_1$. In either case, the forward window of u_i is $F(u_i) \setminus E_i$, and the extension point is a node of $F(u_i)$ that is the greatest in the forward ordering. Note that if $F(u_i) \cap R_r \neq \emptyset$, then it is possible for there to be more than one greatest element of the forward ordering (i.e., a semikernel) in $F(u_i)$.

Now we show how to find the forward extensions of extension points. Suppose that we have some node x that corresponds to a point that is on the convex chain (extension points are all on the convex chain). We wish to find $F(x)$; note that $F(x) \subset N(x)$. We can find the highest-numbered reflex vertex u_i in $N(x)$ (or at least an element of Q_i). The forward window of x is u_i along with the set of nodes of $N(x)$ that are least in the backward ordering. The rest of $F(x)$ is those nodes of $N(x)$ that are equal to x in the forward ordering. The forward extension point of $F(x)$ is again a node of the forward window of x that is greatest in the forward ordering.

2.4. Constructing the Cell Decomposition

Now that we have found the reflex vertices and know how to find forward and backward extensions, we can construct the cell decomposition by following the cell decomposition definition as given in Section 1. Whether two extensions intersect can be determined by testing if the sets of nodes corresponding to these extensions intersect (the node corresponding to the intersection point is the intersection of these sets). The nodes corresponding to a cell of the cell decomposition are those nodes x with $x_1 <_f x <_f x_2$ and $x_3 <_b x <_b x_4$, where x_1, x_2, x_3 , and x_4 are points on the four windows defining the cell as shown in Fig. 12(a). Points on the defining windows in the case that the cell is on the boundary of the polygon are shown in Fig. 12(b) and (c).

Thus, starting from only the point visibility graph $G = PVG(P)$, we have obtained $CD(P)$. This completes the proof of Theorem 2.2.

Fig. 12. Points on the defining windows.

3. The Cell Decomposition Determines the Point Visibility Graph

In this section we show the converse of Theorem 2.2. We then combine our results to obtain necessary and sufficient conditions for spiral polygons to be isomorphic. We also show that isomorphic spiral polygons always have a locality-preserving isomorphism.

Theorem 3.1. *Let $CD(P)$ be the cell decomposition of some spiral polygon P . Then the point visibility graph of P can be determined by examining $CD(P)$.*

The main idea of the proof of this theorem is to use the cell decomposition of P to coordinatize all of the points in P ; the two coordinates given to a point will correspond to the point's position in the backward and forward orderings in the proof of Theorem 2.2. The coordinatization gives a labeling of the nodes in $PVG(P)$. The adjacency of two nodes may then be immediately determined from these labels.

3.1. Coordinatizing P

The first step in assigning coordinates to the points of P is to assign a real number to each point of the convex chain of P . First consider the portion of the convex chain contained in the left end region of P . Along this chain, there will be a series of intersection points with edges of the cell decomposition. Suppose that there are r' such points. Notice that $r' = r$ in the general case, and $r' < r$ if there are *degeneracies* in the cell decomposition (a degeneracy in the cell decomposition arises when some forward extension point is the same as some backward extension point). The convex chain in the left end region will be assigned real values from 0 to 1, beginning with 0 at the vertex v_{n-r} and ending with 1 at the forward extension point of u_0 . The assigned values increase monotonically from 0 to 1 as one proceeds around the boundary in a clockwise fashion. Also, the values are assigned so that the numbers $1/(r' + 1)$, $2/(r' + 1)$, $\dots$, $r'/(r' + 1)$ correspond to the clockwise ordering of the intersection points with the cell decomposition edges (see Fig. 13).

The remainder of the convex chain is assigned values based upon the values assigned above to the left end region. Let x be a point on the convex chain of P , and let $\Phi(x)$

Fig. 13. Assignment of values to the convex chain in the left end region.

denote the value associated with x . Suppose that x is not in the right end region of P , and that y is the forward extension point of x . In this situation, we define $\Phi(y) = \Phi(x) + 1$. Using this definition and the values already assigned for the left end region, we may start with y at the forward extension point of u_0 , and work our way clockwise around the remainder of the convex chain, until y is at u_{r+1} , defining values for all the points along the convex chain as we encounter them (see Figs. 14 and 15). Note that each successive place (clockwise) where a cell decomposition edge meets the convex chain is given a value that is $1/(r' + 1)$ greater than its predecessor. We use the term *cd-values* for these values of $\Phi(\cdot)$ where the cell decomposition edges meet the boundary. Note that the cd-values are completely determined by the cell decomposition.

We may now define coordinates for every point in P . Let p be an arbitrary point in P . Let y denote the backward extension point of p (or u_0 if p has no backward extension) and let x denote the forward extension point of p (or u_{r+1} if p has no forward extension). We define the coordinates of p to be the ordered pair (b, f) (and write $p = (b, f)$) where $b = \Phi(y)$ and $f = \Phi(x)$ (see Fig. 16). Note that all of the points in any semikernel have the same coordinates. Also, any point on the convex chain that is not in an end region has coordinates $(b, b + 2)$ for some b . We have shown two additional points of

Fig. 14. General assignment of values to the convex chain.

Fig. 15. Coordinates of cell decomposition edges.

interest on the convex chain in Fig. 16: x' such that $\Phi(x') = \Phi(x) - 1$ and y' such that $\Phi(y') = \Phi(y) + 1$.

Lemma 3.2. *Two points $p = (b, f)$ and $p' = (b', f')$ are visible iff $f' \geq b + 1$ and $b' \leq f - 1$.*

Proof. The visibility polygon of p is formed by partitioning P with the forward and backward windows of p . Consider Fig. 16, in which the parts of P that do not see p are shaded. If p' does not see p , then it is either because it is backward of the backward extension of p (i.e., in the left shaded region of Fig. 16), or because it is forward of the forward extension of p (i.e., in the right shaded region of Fig. 16). In the former case, $f' < b + 1$, and in the latter, $b' > f - 1$. If the point p' sees p , then it is forward of the backward extension of p and backward of the forward extension of p ; i.e., if it sees p , then $f' \geq b + 1$ and $b' \leq f - 1$. $\square$

Fig. 16. Determining the coordinates of $p = (b, f) = (\Phi(y), \Phi(x))$.

3.2. Classification of Cell Decomposition Elements

We now show that each element of the cell decomposition (cell, edge, or vertex) gives rise to a set of nodes of $PVG(P)$ with specific coordinates. These nodes and their coordinates depend only on the cd-values. Each of the three following sections describe how one type of element maps to a set of nodes.

3.2.1. Vertices. Consider a vertex v of the cell decomposition. This vertex is either a reflex vertex from $u_2, u_3, \dots, u_{r-1}$, on the forward extension of u_0 , on the backward extension of u_{r+1} , or is the intersection of two other extensions. Note that the convex vertices of P are not considered to be vertices of $CD(P)$. As v is a vertex of $CD(P)$, both its forward and backward extensions are part of $CD(P)$, and therefore v 's coordinates are cd-values.

If v is on the forward extension of u_0 (this includes the vertex u_1), and on the backward extension of u_{r+1} (this includes the vertex u_r), then v is a kernel point. In this case, if v is the forward extension point of u_0 , then the kernel is trivial and v gives rise to a single node of $PVG(P)$ with coordinates $(0, \Phi(v_1))$; if v is not the forward extension point of u_0 , then the kernel is nontrivial and v gives rise to $\aleph_1$ nodes of $PVG(P)$ having coordinates $(0, \Phi(v_1))$.

If v is the forward extension point of u_0 , but is not on the backward extension of u_{r+1} , then we consider whether or not v is in the right end region. If it is in the right end region, then it is part of a nontrivial kernel and its corresponding nodes of $PVG(P)$ are produced by the vertex of the cell decomposition that is on the forward extension of u_0 and the backward extension of u_{r+1} , as described above. If it is not in the right end region, then it is part of a trivial semikernel, and gives rise to one node of $PVG(P)$ with the appropriate (cd-valued) coordinates. The case where v is the backward extension point of u_{r+1} but not on the forward extension of u_0 is handled symmetrically.

If v is on the forward extension of u_0 (including u_1 but excluding the forward extension point), but not on the backward extension of u_{r+1} , then v is part of a nontrivial semikernel and gives rise to $\aleph_1$ nodes of $PVG(P)$ having v 's coordinates. The same situation happens if v is on the backward extension of u_{r+1} , but not on the forward extension of u_0 .

In all other cases, v gives rise to a single node of $PVG(P)$.

3.2.2. Edges. Consider now an edge e of the cell decomposition. Such an edge is either an edge of the reflex chain, a decomposition edge in one of the end regions, a decomposition edge not in one of the end regions, or a polygonal chain along the convex chain of the polygon.

If e is an edge of the reflex chain, then consider some point p on e . The forward and backward extensions of p are in $CD(P)$, and therefore p 's coordinates are cd-values. The edge e gives rise to $\aleph_1$ nodes of $PVG(P)$ that all have the same coordinates as p .

If e is an edge in one of the end regions (or in both end regions), then the nodes of $PVG(P)$ that e gives rise to are produced by the vertex where e meets the extension that bounds the end region, unless this vertex is u_1 (or u_r) and e is not the backward window of u_2 (or the forward window of u_{r-1}). In this case let p be a point on e . The point p will have coordinates $(0, x)$ or $(x, \Phi(v_0))$ where x is a cd-value. The edge e gives rise to $\aleph_1$ nodes, each having the same coordinates as p .

If e is a decomposition edge that is not in one of the end regions, then all of the points on e will either have the same b or f coordinate (and this coordinate will be a cd-value). Suppose that the points of e all have the same f coordinate (the other case is symmetric). Then the two endpoints of e are vertices of the cell decomposition with coordinates (b_1, f) and (b_2, f) where $b_1 < b_2$ and both b_1 and b_2 are cd-values. The edge e gives rise to $\aleph_1$ nodes of $PVG(P)$, one for each of the coordinates $\{(b, f) \mid b_1 < b < b_2\}$ where f is the common f coordinate of the points on e .

If e is a polygonal chain along the convex chain of P , then it has two endpoints with coordinates $(b_1, b_1 + 2)$ and $(b_2, b_2 + 2)$, where $b_1 < b_2$. The edge e gives rise to $\aleph_1$ nodes of $PVG(P)$, one for each of the coordinates $\{(b, b + 2) \mid b_1 < b < b_2\}$.

3.2.3. Cells. Finally, we consider a cell c of the cell decomposition. This cell is either in both end regions, in a single end region, has an edge along the convex chain outside of the end regions, or lies outside of the end regions and not along the convex chain. If c is in both end regions, then it is the kernel of P , and its corresponding nodes are produced by one of its vertices.

If c is in a single end region, we assume that it is in the left end region (the case of the right end region is symmetric). All of the points in the cell c will have a b coordinate of 0. Furthermore, the cell c is bounded by two edges having f coordinates f_1 and f_2 with $f_1 < f_2$. The cell c consists of a continuum of nontrivial semikernels, one for each f coordinate between f_1 and f_2 . If we let $F = \{(0, f) \mid f_1 < f < f_2\}$, then c gives rise to $\aleph_1$ nodes for each element of F .

Consider now a cell c that has an edge along the convex chain, and is outside the end regions. There are two vertices $(b_1, b_1 + 2)$, $(b_2, b_2 + 2)$ of $CD(P)$ with $b_1 < b_2$ that are shared by the cell c and the convex chain. One of the cell decomposition edges that bounds c has a constant f coordinate of $b_1 + 2$, and the other has a constant b coordinate of b_2 . The cell gives rise to $\aleph_1$ nodes of $PVG(P)$, one for each of the coordinates in $\{(b, f) \mid b_1 < b < b_2 \text{ and } b_1 + 2 < f < b_2 + 2 \text{ and } b < f - 2\}$ (the last constraint is imposed by the convex chain). Note that $b_1, b_2, b_1 + 2$, and $b_2 + 2$ are all cd-values.

The last element to consider is a cell that is outside the end regions but not along the convex chain. Such a cell is bounded by two edges of a constant b coordinate (with b values b_1 and $b_2, b_1 < b_2$) and two edges of a constant f coordinate (with f values f_1 and $f_2, f_1 < f_2$). Note that if the cell has an edge on the reflex chain, then one of the edges has both a constant b coordinate and a constant f coordinate. This cell gives rise to $\aleph_1$ nodes of $PVG(P)$, one for each coordinate in $\{(b, f) \mid b_1 < b < b_2 \text{ and } f_1 < f < f_2\}$.

3.2.4. Determining the Point Visibility Graph. A cell decomposition completely determines what elements (cells, edges, vertices) it contains, and also determines its cd-values. By the preceding discussion, the elements and cd-values determine the nodes of $PVG(P)$ and the labeling (coordinates) of those nodes. By Lemma 3.2, from the coordinates of two nodes, we can determine whether the nodes are visible; i.e., we can determine if there is an edge of $PVG(P)$ between the two nodes. Thus, $CD(P)$ completely determines $PVG(P)$, and we have completed the proof of Theorem 3.1.

By combining Theorems 2.2 and 3.1, we get:

Theorem 3.3. *Two spiral polygons P and P' are isomorphic iff $CD(P)$ and $CD(P')$ are isomorphic.*

3.3. Homeomorphic Isomorphisms

A one-to-one mapping between two point sets is called *homeomorphic* if both the mapping and its inverse are continuous (i.e., connected sets map to connected sets). It is possible that an isomorphism between two polygons is not homeomorphic; a simple example is shown in Fig. 17. In this example, each cartesian point (x, y) in P is mapped to the point $(2x, y)$ in Q , except that $a = (x_a, y_a)$ is mapped to $a' = (2x_b, y_b)$, and $b = (x_b, y_b)$ is mapped to $b' = (2x_a, y_a)$. This mapping is an isomorphism, as it is one-to-one and the visibility relation is preserved. However, points near a in P do not map to points near a' in Q , so the mapping is not a homeomorphism.

Although it is possible to have nonhomeomorphic isomorphisms, we can show that isomorphic spirals always have a homeomorphic isomorphism:

Theorem 3.4. *If two spiral polygons P and P' are isomorphic, then there is a homeomorphic isomorphism between P and P' .*

Proof. As P and P' are isomorphic, $CD(P)$ and $CD(P')$ are also isomorphic, by Theorem 2.2. As in the proof of Theorem 3.1, we may coordinatize the points of P and P' (we coordinatize so that the cell decomposition isomorphism preserves the associated cd-values—i.e., we use the reflection of P' if necessary). We choose the boundary mappings $\Phi(\cdot)$ to be piecewise linear in the distance around the convex chain. We construct a homeomorphic isomorphism between P and P' by mapping the point (b, f) of P to the point (b, f) of P' . As the boundary mappings $\Phi(\cdot)$ are continuous, this mapping is continuous for the points that have unique coordinates.

Any point that does not have unique coordinates is in a nontrivial semikernel. We consider the types of nontrivial semikernels as described in Lemma 2.1.

If the semikernel is an edge $u_i u_{i+1}$ of the reflex chain, then the points of the semikernel all have coordinates $(b, b + 1)$. We associate with each point p in the semikernel another parameter $\gamma = d(p, u_i)/d(u_{i+1}, u_i)$ where $d(x, y)$ is the Euclidean distance from x to y . This is done in both P and P' . In this semikernel we map the point p of P to

Fig. 17. A nonhomeomorphic isomorphism.

the point of the same semikernel in P' that has the same γ value. This mapping is by itself continuous, and is also continuous with the mapping of the points with unique coordinates.

If the semikernel is a line segment in the left end region, then the points of the semikernel all have coordinates $(0, f)$. The semikernel is a line segment starting at a point c on the convex chain and ending at a point g on the forward window of u_0 (it may be open or closed at its ending point). For such a semikernel, we associate with each point p a parameter $\alpha = d(p, c)/d(g, c)$. The homeomorphic isomorphism then maps a point of this semikernel in P to the point in this semikernel in P' with the same α . As $d(g, c)$ is continuous as a function of α , and the α parameter is itself continuous, this mapping is continuous with itself and with the mapping of the points with unique coordinates.

If the semikernel is a line segment in the right end region, then we associate a β parameter and construct the mapping in a manner symmetric to that done with the α parameter for semikernels of the left end region.

If the semikernel is a nontrivial kernel, then we associate both α and β parameters, as the kernel lies in both the left and right end regions. We then map a point of the kernel of P to the point in the kernel of P' with the same (α, β) pair. This mapping is continuous with the mappings of the line-segment semikernels in the left and right end regions. $\square$

4. Algorithms for Isomorphism of Spiral Polygons

In this section we address the computational complexity of determining if two spiral polygons are isomorphic. We use the Real RAM as our model of computation.

Let u_i be a vertex of the reflex chain of a spiral. We define the *forward extension chain* of u_i as the polygonal chain consisting of u_i , the forward extension $\overline{u_i f}$ of u_i , the forward extension $\overline{f f'}$ of f , the forward extension $\overline{f' f''}$ of f' , and so on until there is no further forward extension. We may similarly define the *backward extension chain* of u_i . Consider now an edge $e_i = u_i u_{i+1}$ in the reflex chain of a spiral. We define the *extension chain* of e_i as the union of the forward extension chain of u_i and the backward extension chain of u_{i+1} (see Fig. 18).

Fig. 18. The extension chain of e_i .

Fig. 19. A narrow orthogonal spiral.

By Theorem 3.3, to tell if two spirals P and P' are isomorphic, we need only compute $CD(P)$ and $CD(P')$ and test these structures for isomorphism. The following lemma establishes a bound on the size of the cell decomposition and thus a lower bound on the time necessary to compute the cell decomposition.

Lemma 4.1. *The cell decomposition of a spiral polygon is of size $O(r^3)$. Furthermore, there are spiral polygons whose cell decomposition is of size $\Omega(r^3)$.*

Proof. As the cell decomposition is planar, we consider counting only the vertices of the cell decomposition and not the edges or cells. We call the region visible to an (open) edge of the reflex chain a *corridor*. Thus, the left and right end regions are both corridors.

We first show that there is a spiral polygon whose cell decomposition is of size $\Omega(r^3)$. Consider an orthogonal spiral having very narrow corridors, as shown in Fig. 19. If the corridors are sufficiently narrow, each edge extension chain will be incident on every reflex vertex, and no two edge extension chains will be identical.

Consider now one corridor, say the region visible to the edge $u_i u_{i+1}$. In this region, each edge extension chain is a two-segment path from u_i to some point on the convex chain to u_{i+1} . Each pair of edge extension chains therefore intersects in one point in this region. As there are $\Omega(r)$ edge extension chains, the cell decomposition will have $\Omega(r^2)$ vertices in this region (see Fig. 20). Furthermore, as the corridors are chosen sufficiently narrow, none of these vertices will be in the intersection of two consecutive corridors.

As we get $\Omega(r^2)$ vertices in every corridor except the two end regions (which contain

Fig. 20. The extension chains intersect in $\Omega(r^2)$ vertices in a corridor.

only $\Omega(r)$ vertices), and there are $\Omega(r)$ such corridors, the cell decomposition of the polygon will have $\Omega(r^3)$ vertices.

To show that the cell decomposition can get no larger than $O(r^3)$, consider again the corridor visible to edge $u_i u_{i+1}$. Any extension chain in this region is a two-segment chain that enters the corridor somewhere on the backward window of u_{i+1} and leaves the corridor somewhere on the forward window of u_i . Thus, each edge extension chain touches the convex boundary (forming a vertex of the cell decomposition) in at most three places in this corridor. Furthermore, as any two-segment chain can intersect another in at most a constant number of places (four), there are at most $O(r^2)$ intersections of extension chains (interior cell decomposition vertices) in this corridor. As there are $O(r)$ corridors, there are at most $O(r^3)$ cell decomposition vertices. (Note that we have overcounted vertices that lie in more than one corridor.) $\square$

Lemma 4.2. *The forward (or backward) extensions of every vertex on the reflex chain in a spiral polygon can be computed in $O(n)$ time.*

Proof. The forward extension points of the vertices $u_0, u_1, \dots, u_{r-1}$ appear in that order in a clockwise traversal of the convex chain of the polygon. Thus, if we walk clockwise around the polygon starting at u_0 , first looking for a point of intersection with $\overline{u_0 u_1}$ (the forward extension point of u_0), then (after finding that point) looking for a point of intersection with $\overline{u_1 u_2}$, and so on, we will find all forward extension points. This walk around the boundary takes $O(n)$ time. $\square$

Lemma 4.3. *The forward extension chains (or backward extension chains) of every vertex on the reflex chain in a spiral polygon can be computed in $O(n + r^2)$ time.*

Proof. First, find all backward extensions of vertices on the reflex chain, in $O(n)$ time, by Lemma 4.2. Insert the backward extension points into the convex chain, splitting the edges where these extension points lie. From the location of these points, it can be determined through which reflex vertex, if any, the forward extension of any point on the convex chain passes (if a point on the convex chain is between the backward extension points of u_i and u_{i+1} , then its forward extension passes through u_i). By an $O(n)$ -time traversal of the convex chain, we can label each of its vertices and edges with which reflex vertex its forward extension passes through.

Next, we find all forward extensions of vertices on the reflex chain, and insert these into the boundary as well. Again, by Lemma 4.2, this takes $O(n)$ time.

The final step of the algorithm is a simultaneous walk of two pointers clockwise along the convex chain. We call the two pointers the *head* and the *tail* pointers. The head and tail both start at the forward extension point q of u_0 . The head starts moving first, and walks clockwise of the tail, searching for the forward extension q' of q . The point q' will be the first intersection of the convex chain with the line through q and the reflex vertex u_i that its forward extension goes through. (The vertex u_i is found in constant time from q by the labeling described above.) When the head reaches q' , it inserts it into the boundary, and then the tail advances to the next forward extension point (either of a reflex vertex or one created where the head stopped). The walk continues in this manner,

alternating between moving the head and the tail, with the head advancing to the forward extension of the tail, and the tail advancing to the next forward extension point in the boundary. The walk ends when the tail reaches the backward extension of v_1 .

To construct the forward extension chains for each vertex of the reflex chain, we start with the chain for each u_i containing the segment from u_i to its forward extension point. Then, during the traversal of the head and tail pointers, as each new extension from p to p' is found, we append the segment $\overline{pp'}$ to whichever extension chain contains the point p .

The two-pointer traversal of the convex chain takes time that is proportional to n plus the number of extension points found. As each forward extension chain has at most $O(r)$ segments (no more than one through each reflex vertex), and there are $O(r)$ chains, the traversal takes $O(n + r^2)$ time. $\square$

Theorem 4.4. *The cell decomposition of a spiral polygon can be computed in $O(n + r^3)$ time.*

Proof. First, we compute the forward and backward extension chains of the vertices of the reflex chain (in $O(n + r^2)$ time, by Lemma 4.3). Consider the arrangement formed by the segments in the forward and backward extension chains. There are $O(r^2)$ segments defining this arrangement, and the arrangement itself has size $O(r^3)$, by Lemma 4.1. We can therefore use the $O(n \log n + k)$ output-size sensitive algorithm of Chazelle and Edelsbrunner [3] to construct this arrangement in $O(r^2 \log r^2 + r^3) = O(r^3)$ time. By traversing the outer face of this arrangement, we can splice on the cell decomposition edges that represent the polygon boundary (edges of the reflex chain or pieces of the convex chain). The total time taken is $O((n + r^2) + r^3) = O(n + r^3)$. $\square$

Thus, the straightforward algorithm to test isomorphism of two spiral polygons will take $O(n + r^3)$ time— $O(n + r^3)$ for computing the cell decompositions, and $O(r^3)$ to test the cell decompositions for isomorphism. Note that cell decomposition isomorphism testing is quite easy, as the reflex chain of one cell decomposition must map to the reflex chain of the other, so there are only two mappings to test (the reflex chain either maps in the same order or maps in the reverse order).

By a closer inspection of the structure of the cell decomposition, we can obtain an $O(n + r^2)$ -time algorithm for spiral polygon isomorphism. This algorithm is based on the idea of the *extension sequence* of a spiral polygon, which is defined as follows.

Consider the extension chain C_i of an edge $e_i = u_i u_{i+1}$ of the reflex chain, and label the points where C_i touches the convex boundary of P with an i , as shown in Fig. 18. If we do this for each extension chain, we end up with a labeling around the convex chain of the spiral polygon, as shown in Fig. 21. The extension sequence is the list of these labels in the order that they occur in a clockwise traversal of the boundary starting at u_0 , and ending just before the next occurrence of the label 0. For Fig. 21, the extension sequence is 0 2 5 1 4 3. The *reflection* of the extension sequence of a polygon P is the extension sequence of a planar reflection of P ; this can be obtained from the original extension sequence by simple string and arithmetic operations.

It is straightforward to show that the cell decomposition is determined by the

Fig. 21. A complete labeling of the convex chain.

extension sequence, and that two cell decompositions that are isomorphic either have the same extension sequence, or have extension sequences that are reflections of each other. Thus, the spiral polygon isomorphism question can be reduced to computing extension sequences and testing if they are equal or reflections. By relatively standard techniques, we can compute the extension sequence of a spiral in $O(n + r^2)$ time. Comparing two extension sequences is trivial and takes $O(r)$ time. We therefore have the following:

Theorem 4.5. *Let P_1 and P_2 be two spiral polygons, each having at most n vertices, in each of which at most r are reflex. Whether P_1 and P_2 are isomorphic can be determined in $O(n + r^2)$ time.*

5. Characterization of Spiral Point Visibility Graphs

In this section we show that the point visibility graphs of spiral polygons are interval graphs. This result was shown for the vertex visibility graph by Everett and Corneil [5]. However, in contrast to the vertex visibility graph result, we are further able to show that no other polygons have point visibility graphs that are interval graphs.

We say that a continuous graph is an *interval graph* if each node of the graph can be associated with an interval $[x_1, x_2]$ of the real line such that two nodes are adjacent in the graph iff the associated intervals intersect. We call a continuous graph *chordal* if it contains no chordless induced cycles of length four or greater. As with finite graphs, continuous interval graphs are a subset of continuous chordal graphs.

Let G be the point visibility graph of some spiral polygon P . Let v be the node of G associated with the point $p \in P$, where p has the coordinates (b, f) . Then we associate the interval $[x_1, x_2] = [b + 1, f]$ with the node v (see Fig. 22).

Lemma 5.1. *Let v and v' be two nodes of G , with associated intervals $[x_1, x_2]$ and $[x'_1, x'_2]$. Then v and v' are adjacent iff the intervals $[x_1, x_2]$ and $[x'_1, x'_2]$ intersect.*

Fig. 22. The interval $[x_1, x_2]$ associated with a point p .

Proof. Let $p = (b, f)$ and $p' = (b', f')$ be the points corresponding to v and v' . Then, by definition, we know that $[x_1, x_2] = [b + 1, f]$ and $[x'_1, x'_2] = [b' + 1, f']$.

Suppose that v and v' are adjacent in G ; i.e., p and p' see each other. By Lemma 3.2, $f' \geq b' + 1$ and $b \leq f' - 1$. Substituting, these inequalities become $x'_2 \geq x_1$ and $x'_1 \leq x_2$. These conditions imply that the two intervals intersect.

Conversely, suppose that the intervals $[x_1, x_2]$ and $[x'_1, x'_2]$ intersect. Then one of the following four cases must occur:

- $x_1 \leq x'_1 \leq x_2 \leq x'_2$.
- $x'_1 \leq x_1 \leq x'_2 \leq x_2$.
- $x_1 \leq x'_1 \leq x'_2 \leq x_2$.
- $x'_1 \leq x_1 \leq x_2 \leq x'_2$.

However, in each case, $x'_2 \geq x_1$ and $x'_1 \leq x_2$. Thus, we know that $f' \geq b + 1$ and $b' \leq f - 1$. Therefore, by Lemma 3.2, p and p' see each other. Thus v and v' are adjacent in G . $\square$

This immediately gives the following lemma:

Lemma 5.2. *If P is a spiral polygon, then $PVG(P)$ is a (continuous) interval graph.*

This lemma implies the following result of Everett and Corneil [5]:

Corollary 5.3. *If P is a spiral polygon, then the vertex visibility graph of P is an interval graph.*

However, in contrast to the situation with vertex visibility graphs, spiral polygons are the only polygons with point visibility graphs that are interval graphs. We can actually show a stronger result:

Lemma 5.4. *If P is a polygon that is not spiral, then $PVG(P)$ is not chordal.*

Proof. Similarly to spiral polygons, we call a maximal sequence of adjacent reflex

Fig. 23. A cycle of length four in $PVG(P)$.

vertices of P a *reflex chain* (note that we do not include the bounding convex vertices here). First, we show that if P is not spiral, then it contains two reflex vertices u and u' that are clearly visible to each other and that are not on the same reflex chain. Consider two reflex vertices x, x' of P that are not on the same reflex chain. Let C be a shortest path from x to x' inside P . It is well known that the interior vertices of the polygonal chain C will be reflex vertices of P [6]. As the endpoints x and x' of C are not on the same reflex chain, there must be at least one edge of C between two reflex vertices u, u' that are not on the same reflex chain. Furthermore, as there are no vertices on C between u and u' , u and u' are clearly visible.

As u and u' are reflex vertices, and are clearly visible, we can find four points p_1, p_2, p_3 , and p_4 such that (1) p_1 and p_2 are in the neighborhood of u , but not visible; (2) p_3 and p_4 are in the neighborhood of u' , but not visible; (3) p_1 sees p_3 and p_4 ; and (4) p_2 sees p_3 and p_4 (see Fig. 23). However, this means that the nodes of $PVG(P)$ corresponding to p_1, p_2, p_3 , and p_4 form a chordless 4-cycle, so $PVG(P)$ is not chordal. $\square$

We summarize the main results of this section in the following theorem:

Theorem 5.5. *For a general polygon P , $PVG(P)$ is chordal iff $PVG(P)$ is an interval graph iff P is a spiral.*

6. Conclusions

We have characterized the point visibility graphs of spiral polygons as interval graphs. We have shown that spiral polygons, their cell decompositions, and their extension sequences all convey the same amount of visibility information, and have given an $O(n^2)$ algorithm for spiral polygon isomorphism based on computing extension sequences. However, we believe that our process of computing extension sequences is not optimal, and a more efficient algorithm is possible:

Conjecture 6.1. *The extension sequence of a spiral polygon can be computed in $o(r^2 + n)$ time.*

The characterizations introduced here can be generalized in a straightforward way to the class of polygons with one hole where both the outer polygon and the hole are convex. In particular, we can show that the point visibility graph of such a polygon is a circular arc graph; this implies the result of Cai and Everett [2] that the vertex visibility graph of such polygons are circular arc graphs.

We have done other work related to that presented here. We have characterized exactly which extension sequences can arise from general or orthogonal spiral polygons [9], and have enumerated them as a function of r . We have also shown that any spiral is isomorphic to a *banana* (a spiral contained in one of the half-planes determined by the line between the two ends of the reflex chain), and that, given any spiral polygon, there is a banana that has the same *vertex* visibility graph.

Another direction that we are investigating is algorithms on extension sequences. If spiral polygons are to be represented as extension sequences, it is natural to see how efficiently properties of the polygon can be computed from the representation. Consider the problem of computing the link-diameter of a spiral from its extension sequence. We can show that the solution to this problem is $|D| + 2$, where $D = \{i \mid i \text{ is before } i - 1 \text{ in the extension sequence}\}$. The set D can be obtained in $O(r)$ time from the extension sequence, and thus we can solve this problem in $O(r)$ time total. Other pure visibility problems on extension sequences have similar solutions.

It is tempting to try to generalize these results to arbitrary simple polygons. However, there are many difficulties associated with such a generalization. The first problem is to settle on a definition of cell decomposition. We have tried several natural cell decomposition definitions, but none of them have the properties that the cell decomposition is finite and that it determines the point visibility graph.

For example, we could extend our definition of the cell decomposition of a spiral polygon to the cell decomposition of a general polygon as follows: let the *zeroth-order cell decomposition* be the polygon itself, and let the *zeroth-order points* of the polygon be the reflex vertices and those convex vertices that are adjacent (along the boundary) to a reflex vertex. We form the *kth-order cell decomposition* from the $(k - 1)$ th-order cell decomposition by adding to it every maximal chord of the polygon that passes through a $(k - 1)$ th-order point and a reflex vertex, provided the reflex vertex is not the end of the chord. The *kth-order points* are then any place where two chords of the *kth-order* cell decomposition intersect, or where one chord ends at the boundary.

However, two polygons can have the same cell decomposition under this definition and yet not have the same point visibility graph. Consider the polygons in Fig. 24, where curved sections represent convex chains with all vertices on the curve. Both of these polygons have the same cell decomposition, as shown in solid lines. The left figure has the property that, given any point on one of the curves, an equilateral triangle can be

Fig. 24. Same cell decomposition, different point visibility graph.

Fig. 25. A polygon with an infinite full cell decomposition.

inscribed so that the vertices of the triangle are on the three curves (with one vertex at the specified point) and the edges of the triangle go through the reflex vertices. This is accomplished by letting the curves be arcs of the circles that are composed of the points p such that the angle rpr' is 60° , where r and r' are the adjacent reflex vertices. This property implies that a *crossing* three-link chain with vertices on the curves and edges on the reflex vertices cannot be inscribed, as is possible in the right figure (shown dashed). By a result of Shermer [14], this means that the point visibility graph of the left figure does not contain an independent set of size three, whereas the right one does; the point visibility graphs of these two figures are therefore nonisomorphic.

Another complicating factor is that it is possible that the cell decomposition for every order is different—there are an infinite number of steps necessary to construct the full cell decomposition. For example, consider the orthogonal polygon shown in Fig. 25. The figure shows part of the k th-order cell decomposition construction, for k from 1 to 4. Only one of the chords of the first cell decomposition is shown, and the chords that it produces in the higher-order cell decompositions. In this polygon the construction process can be carried out indefinitely, as for every two orders of cell decomposition we get a new k th-order point on the top edge of the polygon, further to the left than the previous one. These points will approach, but never reach, the point where the chord through the two reflex vertices intersects the top edge.

These difficulties that we have just discussed lead us to several open questions:

Conjecture 6.2. *There is no finite method of describing a cell decomposition for general polygons so that polygons are isomorphic iff their cell decompositions are.*

Open Question 6.3. Using the definition of cell decomposition above, or any similar one, which polygons have finite full cell decompositions?

Open Question 6.4. Using the definition of cell decomposition above, or any similar one, which polygons have point visibility graphs that are determined by their cell decompositions?

Yet another difficulty with generalizing our techniques is that it is difficult to extract features from the point visibility graph of a general polygon. In our proof, we were easily able to identify the edges of the reflex chain, and the reflex vertices. In general, however, it is not known how to find even a single reflex vertex in a point visibility graph.

Open Question 6.5. Is there a succinct characterization of the vertex equivalence classes (semikernels) of point visibility graphs that contain a node corresponding to a reflex vertex of the polygon?

However, we would like to note here that O'Rourke has suggested that the definition of isomorphism includes the condition that the isomorphism is a homeomorphism [11]. This alternative definition has several advantages, among which are that the boundary of P always maps to the boundary of P' in an isomorphism, and that reflex vertices of P always map to reflex vertices of P' . This eliminates the need for extracting boundary or reflex vertices from point visibility graphs for the purposes of solving the isomorphism question, and gives a starting structure to build an isomorphism from.

Note that for spirals, Theorem 3.4 assures us that our definition and O'Rourke's are equivalent. Furthermore, we conjecture that this is true in general, i.e., Theorem 3.4 holds for all simple polygons:

Conjecture 6.6. *If two simple polygons P and P' are isomorphic, then there is a homeomorphic isomorphism between P and P' .*

Acknowledgments

We would like to acknowledge Joseph O'Rourke for several suggestions, including some of the open problems, and both O'Rourke and an anonymous referee for helpful comments on this paper.

References

1. M. Breen, Clear Visibility and the Dimension of Kernels of Starshaped Sets, *Proc. Amer. Math. Soc.* **85** (1982), 414–418.
2. L. Cai and H. Everett, Visibility Graphs of Polygonal Rings, *Proc. 7th Canadian Conf. on Computational Geometry*, Quebec, 1995, pp. 157–161.
3. B. Chazelle and H. Edelsbrunner, An Optimal Algorithm for Intersecting Line Segments in the Plane, *J. Assoc. Comput. Mach.* **39** (1992), 1–54.
4. J. Culberson and R. Reckhow, Covering Polygons is Hard, *Proc. 29th IEEE Symp. on Foundations of Computer Science*, 1988, pp. 601–611.
5. H. Everett and D. Corneil, Recognizing Visibility Graphs of Spiral Polygons, *J. Algorithms* **11** (1990), 1–26.
6. L. Guibas, J. Hershberger, D. Leven, M. Sharir, and R. Tarjan, Linear-Time Algorithms for Visibility and Shortest Path Problems Inside Triangulated Simple Polygons, *Algorithmica* **2** (1987), 209–233.
7. Y. Ke, An Efficient Algorithm for Link-Distance Problems, *Proc. 5th ACM Symp. on Computational Geometry*, 1989, 69–78.
8. D. Lee and A. Lin, Computational Complexity of Art Gallery Problems, *IEEE Trans. Inform. Theory* **32**, (1986), 276–282.
9. G. MacDonald, Isomorphism and Layout of Spiral Polygons, M.Sc. Thesis, School of Computing Science, Simon Fraser University, Burnaby, 1993.
10. J. O'Rourke, *Art Gallery Theorems and Algorithms*, Oxford University Press, Oxford, 1987.

11. J. O'Rourke, personal communication.
12. T. Shermer, Hiding People in Polygons, *Computing* **42** (1989), 109–131.
13. T. Shermer, Recent Results in Art Galleries, *Proc. IEEE* **80** (1992), 1384–1399.
14. T. Shermer, On Recognizing Unions of Two Convex Polygons and Related Problems, *Pattern Recognition Lett.* **14** (1993), 737–745.
15. S. Suri, Minimum Link Paths in Polygons and Related Problems, Ph.D. Thesis, Johns Hopkins University, Baltimore, MD, 1987.

Received December 1, 1994, and in revised form November 5, 1995.