Detecting and Decomposing Self-Overlapping Curves

Peter W. Shor
Christopher J. Van Wyk

AT&T Bell Laboratories
Murray Hill, New Jersey 07974

ABSTRACT

Paint one side of a rubber disk black and the other side white; stretch the disk any way you wish in three-dimensional space, subject to the condition that from any point in space, if you look down you see either the white side of the disk or nothing at all. Now make the stretched disk transparent but color its boundary black; project its boundary into a plane that lies below the disk. The resulting curve is *self-overlapping*. We show how to test whether a given curve is self-overlapping, and how to count how many essentially different stretchings of the disk could give rise to the same curve.

Some non-simple curves are more non-simple than others. Figure 1(a) shows an elementary non-simple curve. To create this example, begin with a circle in the plane, then insert a finger into its interior and push the boundary around until it crosses itself. This general operation, pushing the boundary from the inside only, repeated a finite number of times, generates the class of *self-overlapping* curves. In particular, it cannot yield either the curve in Figure 1(b), which needs a twist to create it, or the curve in Figure 1(c), which requires two twists or a push from the *outside* to create it.

We define below precisely what it means for a curve to be self-overlapping. Our goal in this paper is to find a way to distinguish self-overlapping curves from other kinds of non-simple curves.* This problem cannot be solved by appeal to the celebrated Whitney-Graustein theorem [W37], which characterizes curves that can be regularly deformed into a circle, since the curve in Figure 1(c) can be deformed regularly into a circle, even though it is not self-overlapping.

DEFINITIONS

For any two points a and b in the plane, $\overline{ab}$ is the line segment between them. Let S^1 and D^2 denote the unit circle and the unit disk, respectively. For a set $S \subset \mathbf{R}^2$ that is homeomorphic to the unit circle S^1, $\tilde{S}$ denotes the union of S and its interior. For a set $S \subset \mathbf{R}^2$ that is homeomorphic to the unit disc D^2, ∂S is the boundary of S and int S is the interior of S. The domain of a mapping f is denoted dom f.

Let $g: S^1 \to \mathbf{R}^2$ be a continuous map of the unit circle into the plane. Then $C = g(S^1)$ is a closed curve in the plane. We use g to keep track of our place when we trace the curve C, which may contain self-intersections. To simplify the notation, we shall write θ instead of $e^{i\theta}$.

We define what it means for curve C to be *self-overlapping* in terms of its decomposition into simple closed curves. Such a decomposition corresponds to a dissection of the unit circle by chords; indeed, we begin the definition of a decomposition with a dissection of the unit circle by a sequence of chords. The following construction tests whether a given decomposition demonstrates that C is self-overlapping.

Let $((\phi_i, \psi_i) \mid 1 \le i \le m)$ be a sequence of non-empty open counterclockwise ranges of angles, and let $\delta_i = \overline{\phi_i \psi_i}$ be the chord of the unit circle determined by ϕ_i and ψ_i. The construction can succeed only if no two chords in the set $\{\delta_i \mid 1 \le i \le m\}$ intersect at a point in int D^2. The sequence of angle ranges gives rise to two sequences of subregions of D^2. Let $\partial\Gamma_0 = S^1$; for $1 \le i \le m$, let $\partial\Delta_{i-1} = \partial\Gamma_{i-1} - (\psi_i, \phi_i)_{\Gamma_{i-1}} \cup \delta_i$ and $\partial\Gamma_i = \partial\Gamma_{i-1} - (\phi_i, \psi_i)_{\Gamma_{i-1}} \cup \delta_i$; finally, let $\partial\Delta_m = \partial\Gamma_m$. (In this definition, the notation $(\phi, \psi)_\Gamma$ means the open interval along $\partial\Gamma$ going counterclockwise from ϕ to ψ.) These subregions are defined only if $\delta_i \subset \Gamma_{i-1}$ for $1 \le i \le m$. Thus, δ_i splits Γ_{i-1} into Δ_{i-1} and Γ_i. The non-intersection condition on the chords implies that Γ_i and Δ_i are homeomorphic to D^2 for all i. Figure 2 shows a dissection of D^2 by a short sequence of angle ranges.

The dissection of D^2 by the sequence of subregions $(\Delta_i \mid 0 \le i \le m)$ gives rise to a decomposition of C in the following way. We describe the decomposition by a sequence of mappings. Let $g_0 = g$; for $1 \le i \le m$, let $x_i = g(\phi_i)$ and $y_i = g(\psi_i)$, and let g_i map $\partial\Delta_{i-1}$ onto $g_{i-1}((\phi_i, \psi_i)_{\Gamma_{i-1}}) \cup \overline{x_i y_i}$ and $\partial\Gamma_i$ onto $g_{i-1}((\psi_i, \phi_i)_{\Gamma_{i-1}}) \cup \overline{x_i y_i}$ such that if $x \in \text{dom } g_{i-1}$, then $g_i(x) = g_{i-1}(x)$. Thus, g_i always maps pieces of S^1 to pieces of C, and for $1 \le j \le i$ it maps the chord δ_j to the jth *diagonal* $\overline{x_j y_j}$. Figure 2 illustrates the development of a short sequence of mappings. For $0 \le i < m$, let $D_i = g_{i+1}(\partial\Delta_i)$; let $D_m = g_m(\partial\Delta_m)$. The ith diagonal chops the closed curve D_{i-1} off of C during the construction.

Definition. The above construction is *valid* if D_i is simple for $0 \le i \le m$.

---

*Bill Thurston conjectures that a planar curve is self-overlapping only if it is the image of the boundary of a homeomorph of the unit disk under an immersion of that homeomorph. We shall not attempt to prove this conjecture.

When a construction is valid, the restriction of g_m to $\partial\Delta_i$ is a homeomorphism for all i.

Let $F:D^2 \rightarrow \mathbf{R}^2$ be a continuous extension of g_m: for $x \in \mathrm{dom}\, g_m$, $F(x) = g_m(x)$, and for $0 \leq i \leq m$, F restricted to Δ_i is a homeomorphism between Δ_i and $\tilde{D}_i$.

Let us pause for a moment to review where in this construction we have made choices. The map $g:S^1 \rightarrow \mathbf{R}^2$ was given. A decomposition of CS is defined by a sequence of angle ranges $((\phi_i, \psi_i) \mid 1 \leq i \leq m)$, which defines a sequence of chords, (δ_i), and two sequences of subregions, (Γ_i) and (Δ_i), of D^2. We constructed a sequence of mappings that culminates in g_m, which defined a sequence of diagonals $(\overline{x_i y_i})$ and a sequence of closed curves (D_i). We chose F as a continuous extension of g_m to D^2. By construction, the restriction of F to S^1 agrees with g, and the restriction of F to $S^1 \cup \{\delta_i \mid 1 \leq i \leq k\}$ agrees with g_k; thus, we can recover the complete status of the construction from the pair (Θ, F), where $\Theta = ((\phi_i, \psi_i) \mid 1 \leq i \leq m)$.

Definition. Let $g:S^1 \rightarrow \mathbf{R}^2$ be given. If there exist Θ and F such that (Θ, F) is a valid construction and for each point $x \in D^2$ there is a neighborhood of x in which F is injective, then g defines a *self-overlapping* curve, and the construction (Θ, F) *demonstrates* this fact.

The dissection Θ and its image under F suggests how we might paste together small pieces of rubber so that together they form a homeomorphic copy of the disk in $\mathbf{R}^3$, the projection of whose boundary into $\mathbf{R}^2$ yields the self-overlapping curve C.

Figure 3(a) shows a self-overlapping curve and a diagonal that suggests how it can be decomposed. Figure 3(b) shows a curve that is not self-overlapping and illustrates why we require that F be locally injective. The diagonal separates C into two simple closed curves, but F is not injective in any neighborhood of the upper endpoint of the diagonal: the interiors of the two closed curves overlap in any neighborhood of that point.

COMPATIBLE DECOMPOSITIONS

For the self-overlapping curves in Figures 1(a), 2, and 3(a), all decompositions are essentially the same, or *compatible*. Figure 4 illustrates that this is not true in general. To explain the difficulty, we define formally what it means for two decompositions to be compatible. Let (Θ, F) and (Θ', F') define two decompositions. The two decompositions are *compatible* if whenever $(\phi', \psi') \in \Theta'$, $F^{-1}(\overline{g(\phi')g(\psi')})$ includes a path that connects ϕ' to ψ' and otherwise lies in $\mathrm{int}\, D^2$, in which case we say that ϕ' and ψ' are *mutually visible* under (Θ, F). We note without proof that compatibility between decompositions of g is an equivalence relation that depends only on Θ and Θ', not on the choice of F and F'; thus we refer below to mutual visibility under Θ alone. We also note that the visibility relation is symmetric and depends only on Θ. Figure 5 shows the inverse image of one of the diagonals in Figure 4(a) with respect to the decomposition in Figure 4(b); the dashed path at g ends at a preimage of d and vice versa. since neither path connects two points in S^1, the two decompositions are incompatible.

ALGORITHMS FOR POLYGONS

In this section we present an algorithm to discover whether a polygon is self-overlapping. This is a key step in the algorithm to solve the problem on general curves.

A curve $C = g(S^1)$ is a polygon if C consists of n line segments or *sides*. In this case, there is an increasing sequence of angles $(\theta_i \mid 0 \leq i < n)$ such that $g(\theta_i)$ is an endpoint of one of the sides of C; the image of these angles under g is the set of vertices of C. A decomposition construction (Θ, F) is a *triangulation* of C if it demonstrates that C is self-overlapping and for $(\phi, \psi) \in \Theta$, $\phi = \theta_j$ and $\psi = \theta_k$ for some j and k. Since triangulations are decompositions, two sequences of angle ranges can define incompatible triangulations.

If there is a triangulation of the polygon then it is certainly self-overlapping. Conversely, Theorem 1 below shows that if (Θ, F) demonstrates that polygon C is self-overlapping then there is a triangulation compatible with (Θ, F). This means that an algorithm to determine whether a polygon is self-overlapping can work by seeking a triangulation of the polygon.

Lemma 1. Let (Θ, F) demonstrate that g defines a self-overlapping polygon P. Let θ be such that $v = g(\theta)$ is a convex vertex of P. There exists an open neighborhood $N_\theta \subset S^1$ of θ such that whenever ϕ and ψ lie on opposite sides of θ in N_θ, ϕ and ψ are mutually visible under Θ.

Proof. Let $N \subset D^2$ be an open neighborhood of θ on which F is injective. The neighborhood $F(N)$ must include a nonempty triangle T two of whose sides coincide with sides of P that are incident to v. Let $N_\theta \subset S^1$ be an open neighborhood of θ such that $g(N_\theta) \subset T$. (See Figure 6.)

Let ϕ, $\psi \in N_\theta$ be two points that lie on opposite sides of θ in N_θ. By construction, $\overline{g(\phi)g(\psi)}$ lies in T. Since F is injective on N and $T \subset N$, $F^{-1}(\overline{g(\phi)g(\psi)})$ includes a path between ϕ and ψ that otherwise lies in the interior of N. Thus, ϕ and ψ are mutually visible under Θ. ∎

Lemma 2. Let (Θ, F) demonstrate that g defines a self-overlapping polygon P. There exist two points θ_1, $\theta_2 \in S^1$ such that $g(\theta_1)$ and $g(\theta_2)$ are vertices of P and θ_1 and θ_2 are mutually visible under Θ.

Proof. Let θ be such that $v = g(\theta)$ is one of the vertices of P with minimum y-coordinate. Since P is self-overlapping, the region $F(D^2)$ can be understood as the "inside" of P. Since v has minimum y-coordinate, all of the inside of P lies above v, so v is a convex vertex. Let ϕ and ψ be the preimages under g of the vertices that precede and follow v on P. If ϕ and ψ are mutually visible under Θ, then we are done.

Otherwise, let $N_\theta = (\phi', \psi')$ be a maximal open neighborhood of θ such that ϕ, ϕ', θ, ψ', and ψ appear in that cyclic order on S^1, and for any ζ and η on opposite sides of θ in N_θ, ζ and η are mutually visible under Θ, as constructed in the proof of Lemma 1 and illustrated in Figure 7. Let $l = \overline{g(\phi')g(\psi')}$. By construction, $F^{-1}(l)$ includes no path that joins ϕ' to ψ' and otherwise lies in $\mathrm{int}\, D^2$. By continuity, however, $F^{-1}(l)$ includes a path Π between ϕ' and ψ' that lies in the inverse image under F of the triangle $\Delta g(\theta)g(\phi')g(\psi')$. Therefore, the path Π intersects S^1 in more than two points. Let χ be a point on the boundary of the intersection of the interior of Π with S^1. Then $F(\chi)$, like $g(\theta)$, is a vertex of P, and θ and χ are mutually visible under Θ. ∎

Theorem 1. Given a decomposition (Θ, F) that demonstrates that polygon P is self-overlapping, there exists a compatible triangulation of P.

Proof. Use Lemma 2 to find points θ_1 and θ_2 that are mutually visible under Θ such that $g(\theta_1)$ and $g(\theta_2)$ are vertices of P. Then $F^{-1}(\overline{g(\theta_1)g(\theta_2)})$ includes a path Π that joins θ_1 to θ_2 and divides D^2 into two simply connected regions Ω_1 and Ω_2. We can modify Θ so it demonstrates that $F(\partial\Omega_1)$ and $F(\partial\Omega_2)$ are both self-overlapping polygons, as follows. We build the two sequences by considering in order the chords defined by Θ. For each, if $(\phi,\psi)\in\Omega$ has both endpoints in $\partial\Omega_1$ or $\partial\Omega_2$, assign it to the modified sequence for the appropriate region. Otherwise, the chord $\overline{\phi\psi}$ intersects Π in a single point (since $F(\overline{\phi\psi})$ and $F(\Pi)$ are both line segments); assign $\overline{\phi\psi}\cap\Omega_1$ to the modified sequence for Ω_1 and $\overline{\phi\psi}\cap\Omega_2$ to the modified sequence for Ω_2.

This proves that a self-overlapping polygon on n vertices can always be decomposed into two self-overlapping polygons, each of which has no more than $n-1$ vertices. Thus, a simple induction suffices to complete the proof of the theorem. ∎

Next we use Theorem 1 to state an algorithm that tests whether a given polygon is self-overlapping.

Algorithm 1. Let $v_0, \ldots, v_{n-1}$ be the vertex sequence of polygon P. For convenience, assume that no three consecutive vertices form a straight angle.

The algorithm uses dynamic programming to find a triangulation of P; it constructs a table $Q_{n\times n}$ where Q_{ij} is one if it is possible to triangulate the $(j-i+1)$-gon whose vertices are $v_i, \ldots, v_j$, and zero otherwise. (All arithmetic on subscripts is carried out modulo n, so if $j<i$, the vertex sequence wraps around from v_{n-1} to v_0, and we treat $j-i+1$ as $j-i+n+1$.) For convenience we set $Q_{i,i+1}=1$.

The first step of the dynamic program is to fill in the values $Q_{i,i+2}$; if v_{i+1} is a convex vertex, then $Q_{i,i+2}=1$, but if v_{i+1} is a reflex vertex then $Q_{i,i+2}=0$. The numbering of the vertices defines an orientation on the polygon, and all triangles for which $Q_{i,i+2}=1$ will be oriented the same way; without loss of generality we assume that this orientation is counterclockwise as one travels from v_i through v_{i+2} to v_{i+1}.

In general, the value of Q_{ij} is one if and only if there exists an index k such that $Q_{ik}=Q_{kj}=1$, $\Delta v_iv_jv_k$ is oriented counterclockwise, v_i, v_j, v_{k+1} and v_{k-1} appear in that order counterclockwise around v_k, and the following four segments do not intersect the interior of $\Delta v_iv_jv_k$: v_iv_{i+1}, $v_{k-1}v_k$, v_kv_{k+1}, and $v_{j-1}v_j$ (see Figure 8). A simple induction shows that when these conditions hold, v_i, v_j, and v_k form a triangle along whose sides we can glue triangulations of v_i through v_k and v_k through v_j so that we can construct an F that is locally injective around v_i, v_j, and v_k. ∎

Since we can compute each element Q_{ij} in $O(n)$ time, Algorithm 1 runs in $O(n^3)$ time. The polygon is self-overlapping if and only if there is an index i such that $Q_{i-1,i}=1$. Therefore we can test in time cubic in the number of vertices whether a polygon is self-overlapping.

To make it possible to reconstruct a triangulation of P from the dynamic program, Algorithm 1 can record at each location Q_{ij} that is set to one a value of k that permitted us to set $Q_{ij}=1$. From these values it is possible to reconstruct a sequence Θ that demonstrates that P is self-overlapping.

GENERALIZATION TO CURVES

In this section we show how to use Algorithm 1 to test whether a curve $C=g(S^1)$ is self-overlapping.

Algorithm 2. For convenience, assume that there is no point $x\in C$ that has more than two preimages under g; if C does contain such multiple crossing points, shift its path slightly to eliminate them. Replace each crossing point and each maximal open interval of C that contains no crossing point by a vertex and connect the vertices to construct a planar graph G whose embedding is topologically the same as C (see Figure 9). Use any of several algorithms ([dFPP88], [G83], [RT85]) to modify the embedding of G so that each edge is replaced by $O(1)$ straight line segments. Let H be the Hamiltonian cycle of a straight-line embedding of G that corresponds to traversing g around S^1; use Algorithm 1 to test whether H is self-overlapping. ∎

If curve C crosses itself at k points, this algorithm runs in $O(k^3)$ time, since the time to construct the dynamic program dominates the time to construct the straight-line embedding of C. This observation could be useful to provide a faster algorithm to determine whether a polygon is self-overlapping. If an n-sided polygon contains k crossing points, and $k<n$, construct the arrangement of the line segments (in $O(n\log n+k)$ time [CE88]), construct the planar graph embedding described in this section (in $O(k)$ time), and run the dynamic program on the resulting polygon (in $O(k^3)$ time).

COUNTING INCOMPATIBLE DECOMPOSITIONS

Let $P=g(S^1)$ be a polygon with n vertices, and let (Θ,F) demonstrate that P is self-overlapping. If the sequence of angle pairs Θ defines a triangulation of P, then for $0\le i\le n-3$, the image $F(\partial\Delta_i)$ is a triangle T_i whose vertices are vertices of P, and $\partial\Delta_i$ contains three values α_i, β_i, γ_i, which are preimages of the vertices of T_i.

The triangulations defined by two sequences of angle ranges, Θ and Θ', are *combinatorially equivalent* if they define the same set of chords in D^2. Thus, combinatorially equivalent triangulations are produced by different orderings of the same set of diagonals. Notice, however, that in general a sequence of angle ranges cannot be reordered arbitrarily, since each diagonal is required to cut off a simple curve.

It is straightforward to modify Algorithm 1 to count the number of combinatorially equivalent ways there are to triangulate P. Instead of setting Q_{ij} to be zero or one, we store in Q_{ij} the number of combinatorially different triangulations of $v_i, \ldots, v_j$. Since a convex n-gon has exponentially many combinatorially different triangulations, all of which are compatible, this count does not tell how many incompatible decompositions there are. Now we shall define some triangulations that have special properties that let them count the incompatible decompositions of a polygon.

A triangulation (Θ,F) of P is a *constrained Delaunay triangulation* (CDT) with respect to a decomposition if for each $0 \le i \le n-3$, there is no value θ such that $g(\theta)$ is a vertex of P, $g(\theta)$ lies inside the circumcircle of T_i, and θ is visible under Θ to all of α_i, β_i, and γ_i. This definition is essentially the same as for simple polygons [LL86], except the notion of visibility is defined with respect to a decomposition.

A triangulation (Θ,F) of P is *locally optimal* if the following is true for every two regions Δ_i and Δ_j that share a chord on their boundaries: Without loss of generality, label the preimages of the vertices so that $\alpha_i = \alpha_j$ and $\gamma_i = \gamma_j$; then β_i does not lie inside the circumcircle of T_j and β_j does not lie inside the circumcircle of T_i. If Δ_i and Δ_j share a chord and do not have this property, then $\Delta\beta_i\beta_j\alpha_i$ and $\Delta\beta_i\beta_j\gamma_i$ do have this property [LL86]. From this it is clear that a locally optimal triangulation of a polygon P always exists, and that a constrained Delaunay triangulation is locally optimal.

The next theorem shows that a locally optimal triangulation is a constrained Delaunay triangulation, which proves that constrained Delaunay triangulations exist, and also that we can compute them relatively easily.

Theorem 2. Suppose that $P = g(S^1)$ has no four cocircular vertices and Θ defines a locally optimal triangulation of P. Then Θ defines a CDT of P.

Proof. The proof is a modification of the proof for simple polygons [LL86], and is omitted in this abstract.

If no four of the vertices of a simple polygon are cocircular, then the constrained Delaunay triangulation is unique [LL86]. We shall prove a stronger result.

Theorem 3. Suppose that $P = g(S^1)$ has no four cocircular vertices. Two decompositions Θ and Θ' of P have combinatorially equivalent constrained Delaunay triangulations if and only if they are compatible.

Proof. Omitted.

Theorem 3 implies that when P has no four cocircular vertices, we can count the number of incompatible decompositions it has by finding the number of combinatorially inequivalent constrained Delaunay triangulations. Algorithm 3 is a modification of Algorithm 1 that does this. It fills a table $Q_{n\times n\times n}$ by setting Q_{ijk} to be the number of combinatorially different locally optimal triangulations of the $(j-i+1)$-gon whose vertices are $v_i, \ldots, v_j$ that include $\Delta v_i v_j v_k$. Obviously, $Q_{ijk} = 0$ when k does not follow i and precede j in cyclic order.

Algorithm 3. The first step of the dynamic program sets $Q_{i,i+2,i+1} = 1$ if and only if v_{i+1} is a convex vertex. The general step of the dynamic program sets Q_{ijk} to $(\sum_a Q_{ika})\times(\sum_b Q_{kjb})$ where the summation indices a and b are such that $Q_{ika} > 0$, $Q_{kjb} > 0$, $\Delta v_i v_j v_k$ is oriented counterclockwise and obeys the local optimality property with respect to both triangles $\Delta v_i v_a v_k$ and $\Delta v_k v_b v_j$, the vertices v_i, v_j, v_{k+1} and v_{k-1} appear in that order counterclockwise around v_k, and the following four segments do not intersect the interior of $\Delta v_i v_j v_k$: $v_i v_{i+1}$, $v_{k-1} v_k$, $v_k v_{k+1}$, and $v_{j-1} v_j$. Thus, v_i, v_j, and v_k form a triangle along whose sides we can glue locally optimal triangulations of v_i through v_k and v_k through v_j to derive a locally optimal triangulation of v_i through v_j. ■

Since the range of values of a and b that must be considered to compute Q_{ijk} do not overlap, it is easy to compute Q_{ijk} in $O(n)$ time, which leads to a running time of $O(n^4)$ for Algorithm 3. If instead of a simple three-dimensional table we maintain a matrix $Q_{n\times n}$ of sorted sequences, where Q_{ij} contains the partial sums of the values of Q_{ijk}, sorted by increasing angle at v_k in $\Delta v_i v_j v_k$, then we can reduce this running time to $O(n^3 \log n)$.

If a polygon contains four cocircular vertices that are mutually visible under some decomposition, then it has combinatorially different Delaunay triangulations that are compatible. Since Algorithm 3 counts combinatorially different Delaunay triangulations, it will not count correctly the number of incompatible decompositions of the polygon. To prevent this, modify the dynamic program so that if T_{ijk_1} and T_{ijk_2} are to be set to the same value because v_i, v_j, v_{k_1}, and v_{k_2} are cocircular and mutually visible, then only $T_{i,j,\min\{k_1,k_2\}}$ is set to this value, while $T_{i,j,\max\{k_1,k_2\}}$ is set to zero.

If we use Algorithm 3 instead of Algorithm 1 as a step in Algorithm 2, then we can count the number of incompatible decompositions of any curve.

OPEN PROBLEMS

Can the detection problem be solved in sub-cubic time?

Let C be a self-overlapping plane curve and let R be an open region in $\mathbf{R}^2$–C. Under any mapping F defined by a decomposition of C, every point in R has the same number of preimages under F (this follows from theorems about winding numbers [A66]); thus we can speak of the number of *layers* that cover R, independent of the mapping F. We say that C is a k-*layer* curve if k is the maximum value such that no region in $\mathbf{R}^2$–C is covered by more than k layers; for example, the curve in Figure 4 is a three-layer curve. We know of no two-layer curve that has two incompatible decompositions, and conjecture that none exists.

The curve in Figure 4 has two "holes"; for any k, it can be generalized to have k holes, $k+1$ layers, and k incompatible decompositions. It can also be glued to several copies of itself to form a curve that has $2k$ holes, 3 layers, and 2^k incompatible decompositions. These observations lead us to conjecture that in general the number of incompatible decompositions of a k-layer curve C is not divisible by a prime larger than k.

Is there a constructive way to define the class of self-overlapping curves when there are infinitely many self-overlaps? It is not enough to take $m = \infty$ in the definition of this paper, because a sequence of diagonals could approach a limit and not decompose the whole region into simple curves; the region "Δ_∞" could be non-self-overlapping.

ACKNOWLEDGMENTS

Micha Sharir posed this problem at the Eighth Geometry Day at the Courant Institute. Chi-Yuan Lo suggests that its solution could be useful in processing VLSI layouts that are represented by polygon boundaries [NS80]. We are grateful for several discussions with Ken Clarkson, Subhash Suri, and Bob Tarjan. Steve Fortune offered some suggestions that greatly simplified the notation (Really!). David Johnson, David Levine, and Alex Schaffer made helpful comments on the manuscript.

REFERENCES

[A66] L. V. Ahlfors, *Complex Analysis*, 2d ed., 1966, New York: McGraw-Hill, 114-118.

[CE88] B. Chazelle and H. Edelsbrunner, "An optimal algorithm for intersecting line segments in the plane," *Proc. 29th Ann. Symp. on Found. of Comput. Sci.*, 1988, 590-600.

[dFPP88] H. de Fraysseix, J. Pach, and R. Pollack, "Small sets supporting Fary embeddings of planar graphs," *Proc. 20th Ann. Symp. on Th. Comput.*, 1988, 426-433.

[G83] D. H. Greene, "Efficient coding and drawing of planar graphs," 1983, unpublished manuscript.

[LL86] D. T. Lee and A. K. Lin, "Generalized Delaunay triangulation for planar graphs," *Discrete and Comput. Geom.* 1 (1986), 201-217.

[NS80] M. E. Newell and C. H. Sequin, "The inside story on self-intersecting polygons," *Lambda* 1 (1980), 20-21.

[RT85] P. Rosenstiehl and R. E. Tarjan, "Rectilinear planar layouts and bipolar orientations of planar graphs," *Discrete and Comput. Geom.* 1 (1986), 343-353.

[W37] H. Whitney, "On regular closed curves in the plane," *Compositio Math.* 4 (1937), 276-284.

FIGURE CAPTIONS

1. The curve in (a) is a self-overlapping curve. Those in (b) and (c) are not.

2. The left column shows the dissection of D^2 by the sequence $((\phi_1,\psi_1),(\phi_2,\psi_2))$. The right column shows the image of D^2 together with appropriate chords as the sequence of mappings g_0, g_1, and g_2 is produced. The shaded regions indicate int D_0 and int D_1.

3. The diagonal in (a) shows that the curve is self-overlapping. The diagonal in (b), however, does not yield a proper decomposition of the non-self-overlapping curve shown.

4. This figure has two incompatible decompositions. The diagonals in one are $\overline{af}$ and $\overline{be}$; the diagonals in the other are $\overline{ch}$ and $\overline{dg}$.

5. The points in S^1 that are preimages under g of the labelled points in Figure 4 are labelled with the corresponding letters. The chords show that we are using the decomposition in which $\overline{af}$ and $\overline{be}$ are diagonals. Under a mapping F defined by this decomposition, the inverse image of $\overline{dg}$ does not include a path between points on S^1.

6. Figure (a) depicts most of the notation in the proof of Lemma 1. Figure (b) shows the images of the neighborhoods N and N_θ under F and g, respectively, as well as triangle T.

7. This illustration for the proof of Theorem 1 uses a simple polygon. Thus, F and g are homeomorphisms, and we need only draw the situation in the plane that contains P. The figure shows the image of one possible choice of maximal open neighborhood N_θ.

8. General step of the dynamic program.

9. The transformation from curve to straight-line Hamiltonian planar graph. The edges of the two-cycle have been bowed out to make both visible.

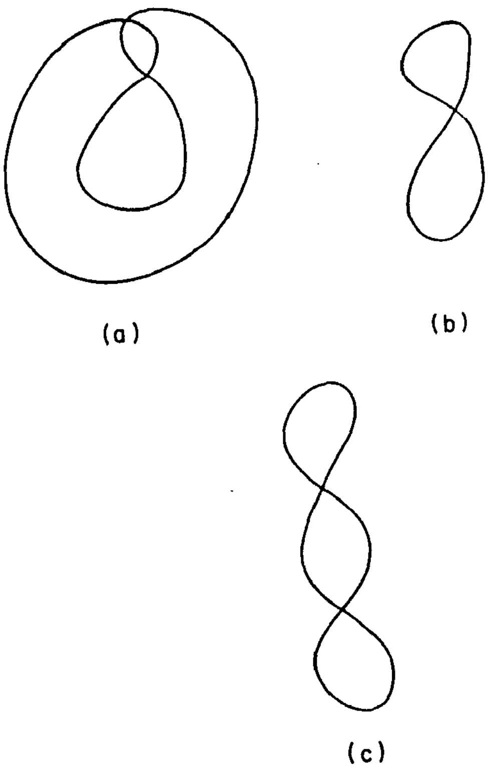


FIG. 1

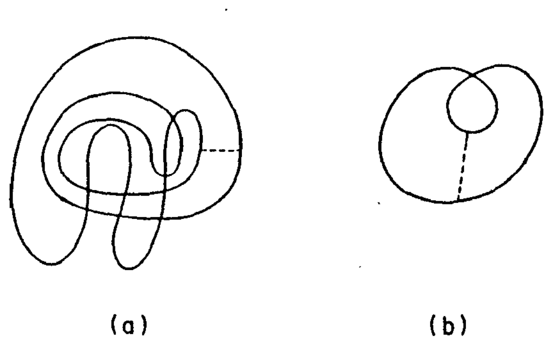


FIG. 3

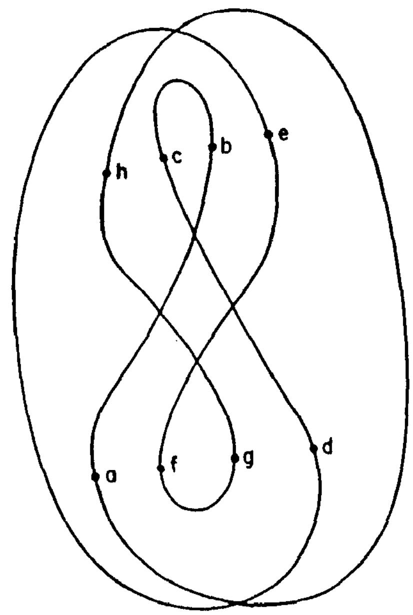


FIG. 4

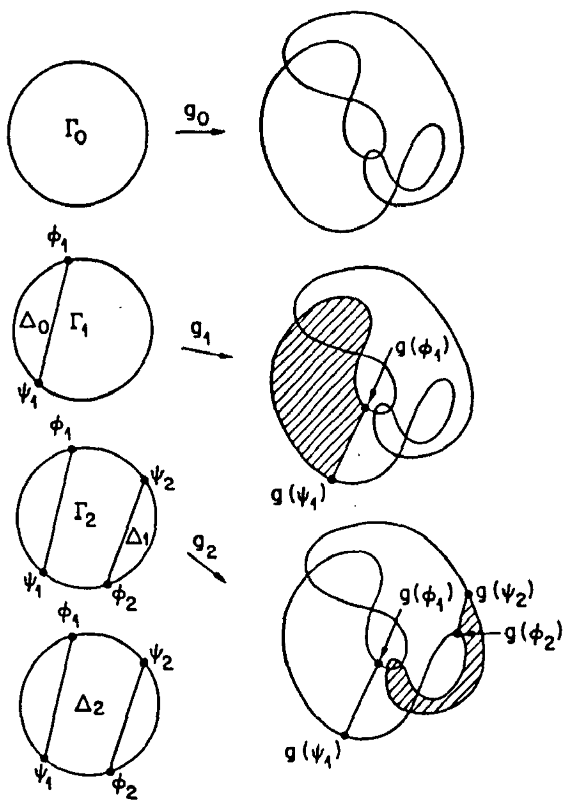


FIG. 2

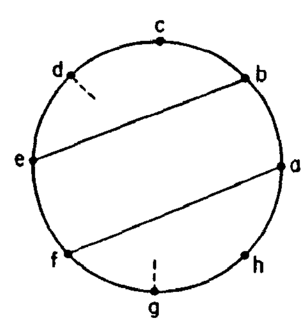
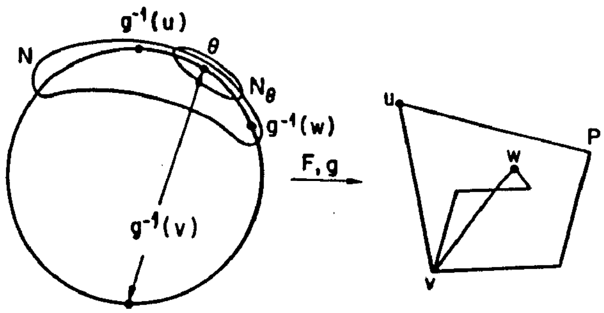
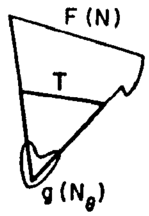


FIG. 5



(a)



(b)

FIG. 6

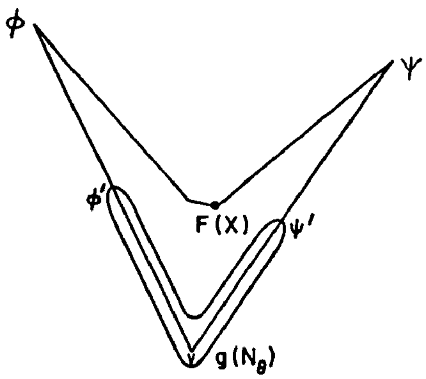


FIG. 7

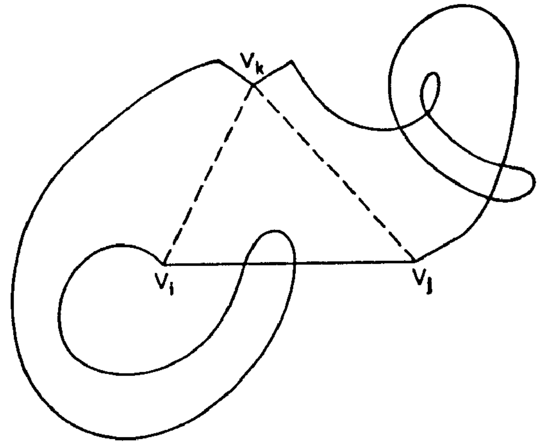


FIG. 8

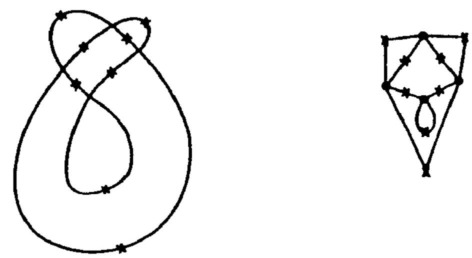


FIG. 9