One-Dimensional Computational Topology Notes

Jeff Erickson

I am attempting to write at least a few pages of notes for each lecture in the Fall 2020 iteration of this course. By default, you should expect these notes to be rough drafts, with lots of missing detail; don't let the long-windedness of the early notes fool you! I can't promise that the notes are 100% accurate reflections of the actual lectures; for that, you should watch the actual lecture videos.'

I'm *attempting* to write each note to exactly cover one 75-minute lecture, to force myself to prioritize the most fundamental results, sometimes at the expense of the technical state of the art, prerequisite material (like point-set topology or dynamic-forest data structures), and historical anecdotes.

Similarly, I'm writing these notes in Markdown instead of LaTeX, in part to intentionally focus my time on *writing* instead of typographical tricks. As a result, the notes are typographically boring/ugly (at least until I have time to figure out how to use Pandoc templates and filters). Comments on the course Campuswire page are welcome. For similar reasons, the notes are embarrassingly short on references; I'll try to provide direct links instead of the usual bibliography.

Eventually I'll make the source files available on Github to attract bug reports, feature requests, and pull requests. Stay tuned.

---

1 Simple Polygons

The Jordan Curve Theorem and its generalizations are the formal foundations of many results, if not *every* result, in two-dimensional topology. In its simplest form, the theorem states that any simple closed curve partitions the plane into two connected subsets, exactly one of which is bounded. Although this statement is intuitively clear, perhaps even obvious, the generality of the term "simple closed curve" makes a formal proof of the theorem incredibly challenging. A complete proof must work not only for sane curves like circles and polygons, but also for more exotic beasts like fractals and space-filling curves. Fortunately, these exotic curves rarely occur in practice, except as counterexamples in point-set topology textbooks.

A full proof of the Jordan Curve Theorem requires machinery that we won't cover in this class (either point-set topology or singular homology). Here I'll consider only one important special case: *simple polygons*. Polygons are by far the most common type of closed curve employed in practice, so this special case has immediate practical consequences.

Most published proofs of the full Jordan Curve Theorem both dismiss this special case as trivial and rely on it as a key lemma. Indeed, the proof is ultimately *elementary*. Nevertheless, the lemma and its proof are the origin of several fundamental algorithmic tools in computational geometry and topology.

1.1 Definitions

A *path* in the plane is an arbitrary continuous function $\pi\colon [0,1] \to \mathbb{R}^2$, where $[0,1]$ is the unit interval on the real line. The points $\pi(0)$ and $\pi(1)$ are the *endpoints* of the path. A *closed curve* (or *cycle*) in the plane is a continuous function from the unit circle $S^1 = \{(x,y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\}$ to the plane.

A path or cycle is *simple* if it is injective, or intuitively, if it does not "self-intersect". To avoid excessive formality, we do not normally distinguish between a simple path or cycle (formally a function) and its image (a subset of the plane).

A subset X of the plane is *(path-)connected* if there is a path in X from any point in X to any other point in X. A *(path-)component* of X is a maximal path-connected subset of X.

Theorem (The Jordan Curve Theorem). *The complement $\mathbb{R}^2 \setminus C$ of any simple closed curve C in the plane has exactly two components.*

A *polygonal chain* is a path that passes through a sequence of points $p_0, p_1, \ldots, p_n$, where each subpath from p_{i-1} to p_i is the straight line segment $p_{i-1}p_i$. The points p_i are the *vertices* of the polygonal chain, and the segments $p_{i-1}p_i$ are its *edges*. We usually assume without loss of generality that no pair of consecutive edges is collinear, and in particular, that no two consecutive vertices coincide.

A polygonal chain is *closed* if it has at least one edge and its first and last vertices coincide (that is, if $p_0 = p_n$) and *open* otherwise. Closed polygonal chains are also called *polygons*; a polygon with n vertices and n edges is also called an *n-gon*. We can regard any polygon as a closed curve in the plane. Every simple polygon has at least three vertices.

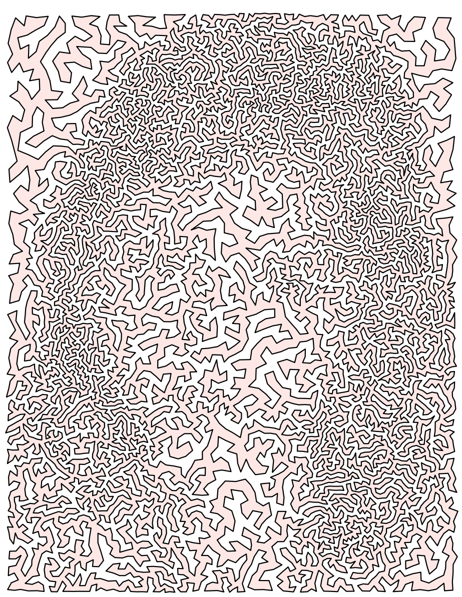

Figure 1: A simple 10000-gon, with interior shaded

Theorem (The Jordan Polygon Theorem). *The complement $\mathbb{R}^2 \setminus P$ of any simple polygon P in the plane has exactly two components.*

Even though this special case of the Jordan Curve Theorem considers only polygonal curves, the definition of "connected" allows for *arbitrary* paths between points.

1.2 Proof of the Jordan Polygon Theorem

Fix a simple polygon P with n vertices. Without loss of generality, assume no two vertices pf P have equal x-coordinates. The vertical lines through the vertices partition the plane into $n+1$ *slabs*, two of which (the leftmost and rightmost) are actually halfplanes. The edges of P subdivide each slab into a finite number of regions we call *trapezoids*, even though some of these regions are actually triangles, and others are unbounded in one or more directions.

(Figure!)

The boundary of each trapezoid consists of (at most) four line segments: the *floor* and *ceiling*, which are segments of polygon edges, and the *left* and *right walls*, which are segments of the vertical slab boundaries. The endpoints of each vertical wall (if any) lie on the polygon P.

Formally, we define each trapezoid to include its walls but not its floor, its ceiling, or any vertex on its walls. Thus, each trapezoid is connected, any two trapezoids intersect in a common wall or not at all, and the union of all the trapezoids is $\mathbb{R}^2 \setminus P$. In particular, a trapezoid is a convex (and therefore connected) region in the plane, not a polygon!

Lemma $\leq$ 2. $\mathbb{R}^2 \setminus P$ *has at most two components.*

Proof: Direct the edges of P in increasing index order (modulo n). Informally, we label every trapezoid *left* or *right* depending on whether a person walking around P would see that trapezoid immediately to their left or immediately to their right. More formally, we label every trapezoid that satisfies at least one of the following conditions *left*:

- The ceiling is directed from right to left.
- The floor is directed from left to right.
- The right wall contains a vertex p_i, and the incoming edge $p_{i-1}p_i$ is below the outgoing edge $p_i p_{i+1}$
- The left wall contains a vertex p_i, and the incoming edge $p_{i-1}p_i$ is above the outgoing edge $p_i p_{i+1}$

These conditions apply verbatim to unbounded and degenerate trapezoids. There are four symmetric conditions for labeling a trapezoid *right*. Every trapezoid is labeled left or right *or (as far as we know) possibly both*.

Now imagine imagine walking once around the polygon, facing directly forward along edges and turning at vertices, and consider the sequence of trapezoids immediately to our left, as suggested by the white arrows in the figure below. Without loss of generality, start at the leftmost vertex p_0. Whenever we traverse a directed edge $p_{i-1}p_i$ from right to left, our left hand sweeps through all trapezoids immediately below that edge. Whenever we reach a vertex p_i whose neighbors are both to the right of p_i, where the edges make a right (clockwise turn), our hand sweeps through the trapezoid just to the left of p_i. The other cases are symmetric. The resulting sequence of trapezoids contains every left trapezoid at least once (and at most four times); moreover, any adjacent pair of trapezoids in this sequence share a wall and thus have a connected union. So the union of the left trapezoids is connected.

A symmetric argument implies that the union of the right trapezoids is also connected, which completes the proof.

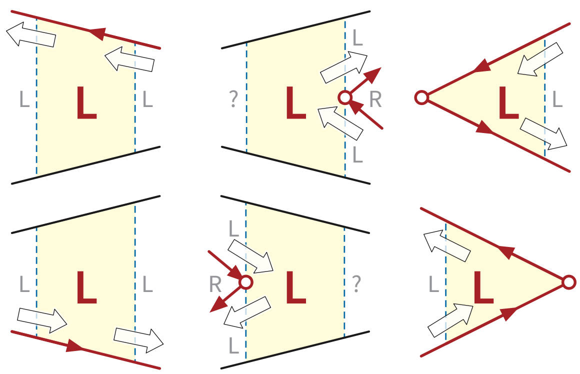

Figure 2: Left trapezoids

Lemma ≥ 2. *$\mathbb{R}^2 \setminus P$ has at least two components.*

Proof (Jordan): Label each trapezoid *even* or *odd* depending on the parity of the number of polygon edges directly above the trapezoid. Thus, within each slab, the highest trapezoid is even, and the trapezoids alternate between even and odd.

Consider two trapezoids A and B that share a common wall, with A on the left and B on the right, on the vertical line ℓ through vertex p_i. If vertices p_{i-1} and p_{i+1} are on opposite sides of ℓ, exactly the same number of polygon edges are above A and above B. Suppose p_{i-1} and p_{i+1} lie to the left of ℓ. If p_i lies below the wall $A \cap B$, then A and B are below the same number of edges; otherwise, A is below two more edges than B. Similar cases arise when p_{i-1} and p_{i+1} both lie to the right of ℓ. In all cases, A and B have the same parity.

It follows by induction that any two trapezoids in the same component of $\mathbb{R}^2 \setminus P$ have the same parity, which completes the proof.

The Jordan Polygon Theorem follows immediately from Lemmas ≤ 2 and ≥ 2. In particular, if the polygon is oriented counterclockwise (the way god intended), then “right” and “even” mean “outside”, and “left” and “odd” mean “inside”.

1.3 Point-in-Polygon Test

The proof of Lemma ≥ 2 immediately suggests a standard algorithm to test whether a point is inside a simple polygon in the plane in linear time: Shoot an arbitrary ray from the query point, count the number of times this ray crosses the polygon, and return TRUE if and only if this number is odd. This algorithm appears in Gauss’ notes (written around 1830 but only published after his death); it has been rediscovered many times since then.

To make the ray-parity algorithm concrete, we need one numerical primitive from computational geometry. A triple (q, r, s) of points in the plane is *oriented counterclockwise* if walking from q to r and then to s requires a left turn, or *oriented clockwise* if the walk requires a right turn. More explicitly, consider the 3×3 determinant

$$\Delta(q,r,s) = \det \begin{bmatrix} 1 & q.x & q.y \\ 1 & r.x & r.y \\ 1 & s.x & s.y \end{bmatrix} = (r.x - q.x)(s.y - q.y) - (r.y - q.y)(s.x - q.x).$$

The triple (q,r,s) is oriented counterclockwise if $\Delta(q,r,s) > 0$, oriented clockwise if $\Delta(q,r,s) < 0$, and collinear if $\Delta(q,r,s) = 0$. (The absolute value of $\Delta(q,r,s)$ is twice the area of the triangle $\triangle qrs$.)

Straightforward case analysis implies that the vertical ray from q crosses the line segment rs if and only if q lies between the vertical lines through r and s, and $\Delta(q,r,s)$ has the same sign as $r.x - s.x$.

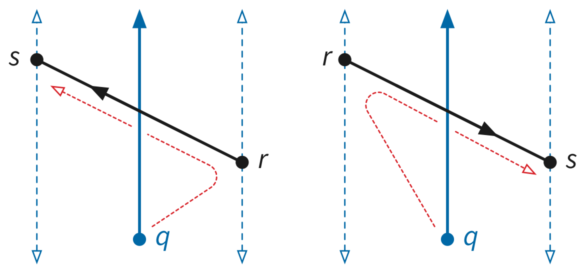

Figure 3: Ray-crossing test

Finally, here is the algorithm in (pseudo)Python. The input polygon P is represented by an array of consecutive vertices, which are assumed to be distinct. The algorithm returns $+1$, -1, or 0 to indicate that the query point q lies inside, outside, or directly on P, respectively. To correctly handle ties between x-coordinates, the algorithm treats any polygon vertex on the vertical line through q as though it were slightly to the left. The algorithm clearly runs in $O(n)$ time.

```
def PtInPolygon(P, q):
    sign = -1                       // outside if no crossings
    n = size(P)
    for i in range(n):
        r = P[i]
        s = P[(i+1)% n]
        Delta = (r.x - q.x)*(s.y - q.y) - (r.y - q.y)*(s.x - q.x)
        if s.x <= q.x < r.x         // positive crossing?
            if Delta > 0:
                sign = -sign
            elif Delta == 0:
                return 0
        elif r.x <= q.x < s.x       // negative crossing?
            if Delta < 0:
                sign = -sign
            elif Delta == 0:
                return 0
    return sign
```

1.4 Polygons Can Be Triangulated

Most algorithms that operate on simple polygons actually require a decomposition of the polygon into simple pieces that are easier to manage. In the most natural decomposition, the pieces are triangles that meet edge-to-edge. More formally, a *triangulation* is a triple of sets (V, E, T) with the following properties.

- T is a finite set of triangles in the plane with disjoint interiors.
- E is the set of edges of triangles in T.
- Any two segments in E have disjoint interiors.
- V is the set of vertices of triangles in T.

The third condition guarantees that the intersection of any two triangles in T is either an edge of both, a vertex of both, or empty. If the union of the triangles in T is equal to the closure of the interior of a simple polygon P, we call $(V.E.T)$ a *triangulation of* P.

If moreover V is the set of vertices of P, then $(V.E.T)$ a *frugal* triangulation of P. Every edge of a frugal triangulation is either an edge of P or an *(interior) diagonal*, meaning a line segment between two vertices of P and that otherwise lies in the interior of P.

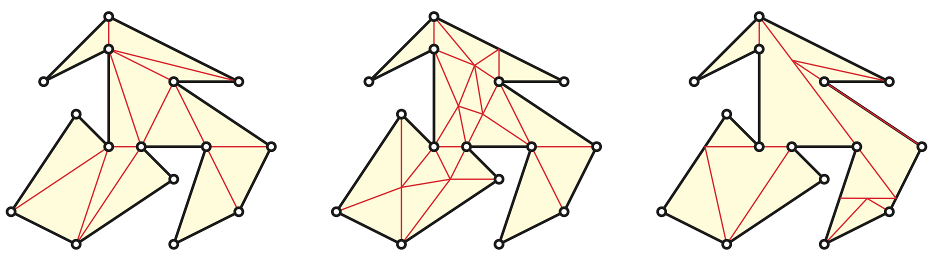

Figure 4: A frugal triangulation, a non-frugal triangulation, and a non-triangulation of a simple polygon

After playing with a few examples, it may seem obvious that every simple polygon has a frugal triangulation, but a formal proof of this fact is surprisingly subtle; several incorrect (or at least incomplete) proofs appear in the literature. The first complete, correct, axiomatic proofs were developed by Dehn (1899, unpublished) and Lennes (1911), although some components of their arguments already appear in the Gauss's posthumously published notes.

The following proof is somewhat more complicated (and intentionally *less* formal!) than Dehn's and Lennes's arguments, but it directly motivates a faster algorithm for constructing diagonals.

Diagonal Lemma (Dehn, Lennes): *Every simple polygon with at least four vertices has an interior diagonal.*

Proof: Let P be a simple polygon with vertices $p_0, p_1, \ldots, p_{n-1}$ for some $n \geq 4$. As in our previous proof, we assume without loss of generality that no two vertices of P lie on a common vertical line. We begin by subdividing the closed interior of P into trapezoids with vertical line *segments* through the vertices. Specifically, for each vertex p_i, we cut along the longest vertical segment through p_i in the closure of the interior of $\sim P$. The resulting subdivision, which is called a *trapezoidal decomposition* of P, can also be obtained from the slab decomposition we used to prove the Jordan polygon theorem by removing every exterior wall and every wall that does not end at a vertex of P.

Every trapezoid in the decomposition has exactly two polygon vertices on its boundary. Call a trapezoid *boring* if the line segment between these two vertices cuts through the interior of the trapezoid, and therefore is a diagonal of P, and *interesting* otherwise. Every interesting trapezoid either has two vertices of P on its ceiling, or two vertices of P on its floor.

If any of the trapezoids is boring, we immediately have a diagonal. Yawn.

Any path through the interior of P that starts in a ceiling trapezoid and ends in a floor trapezoid must pass through a boring trapezoid. So if every trapezoid is interesting, then every trapezoid is interesting *the same way*—either every trapezoid has two vertices on its ceiling, or every trapezoid has two vertices on its floor. Thus, P is a special type of polygon we call a *monotone mountain*: any vertical line intersects at most two edges of P, and the leftmost and rightmost vertices of P are connected by a single edge of P.

Without loss of generality, suppose p_0 is the leftmost vertex, p_{n-1} is the rightmost vertex, and every other vertex is above the edge $p_0 p_{n-1}$ (so every trapezoid has two vertices on its ceiling). Call a vertex p_i *convex* if the interior angle at that vertex is less than π, or equivalently, if the triple (p_{i-1}, p_i, p_{i+1}) is oriented *clockwise*. Every monotone mountain has at least one convex vertex p_i other than p_0 and p_{n-1}; take, for example, the vertex furthest above the floor $p_0 p_{n-1}$. For any such vertex p_i, the line segment $p_{i-1} p_{i+1}$ is a diagonal.

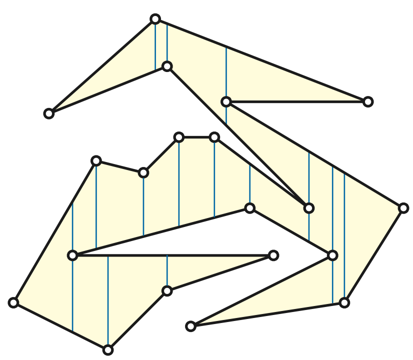

Figure 5: A trapezoidal decomposition

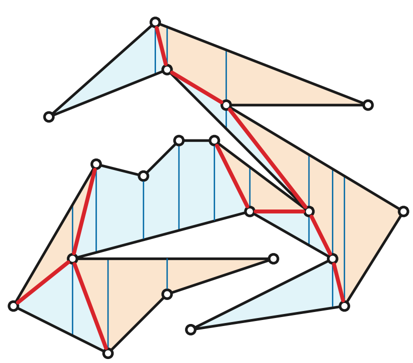

Figure 6: Decomposing a polygon into monotone mountains along boring diagonals

Triangulation Theorem: *Every simple polygon has a frugal triangulation.*

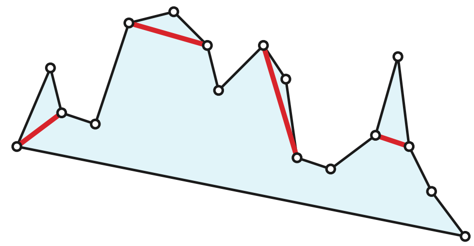

Figure 7: Four diagonals in a monotone mountain

Proof (Dehn, Lennes): The theorem follows by induction from the previous lemma. Intuitively, to triangulated any polygon, we can split any polygon along a diagonal, and then recursively triangulate each of the two resulting smaller polygons.

Let P be a simple polygon with n vertices $p_0, p_1, p_2, \ldots, p_{n-1}$. If P is a triangle, it has a trivial triangulation, so assume $n > 3$. Suppose without loss of generality (reindexing the vertices if necessary) that $d = p_0 p_i$ is a diagonal of P, for some index i. Let P^+ and P^- denote the polygons with vertices $p_0, p_i, p_{i+1}, \ldots, p_{n-1}$ and $p_0, p_1, p_2, \ldots, p_i$, respectively. The definition of "diagonal" implies that both P^+ and P^- are simple. Color each edge of P *red* if it is an edge of P^+ and *blue* otherwise; every blue edge is an edge of P^-.

Let q be any point in the interior of P, but not on the diagonal d, and let R be any ray starting at q. The definition of "interior" implies that R crosses an odd number of edges of P. Thus, either R crosses the boundary of P^+ an odd number of times (possibly including one crossing of d) and crosses the boundary of P^- an even number of times (possibly including one crossing of $p_0 p_i$), or vice versa. We conclude that the interior of P^+, the interior of P^-, and the diagonal d *cover* the interior of P.

Now here's the subtle part that most proofs omit. Let U be any open disk in the interior of P that intersects $p_0 p_i$; such a disk exists because d is an *interior* diagonal. (We had to use that fact somewhere!) The set $U \setminus p_0 p_i$ has exactly two components. (We are *not* invoking the Jordan curve theorem here, but rather a much more basic fact called *Pasch's axiom*.) Choose arbitrarily points q^+ and q^-, one in each component. Let R^+ and R^- be parallel rays starting at q^+ and q^-, respectively, such that R^+ contains R^-. Then R^+ crosses d but R^- does not, and R^+ and R^- cross exactly the same edges of P.

Figure!!

As above, R^+ (and therefore R^-) crosses an odd number of edges of P. Without loss of generality, suppose R^+ (and therefore R^-) crosses an even number of red edges and an odd number of blue edges. Then, because R^+ crosses d, the point q^+ lies inside P^+ and outside P^-. Similarly, q^- lies inside P^- and outside P^+, because R^- does not cross d. We conclude (finally!) that the interiors of P^+ and P^- are disjoint subsets of the interior of P.

The inductive hypothesis implies that P^+ has a frugal triangulation (V^+, E^+, T^+) and that P^- has a frugal triangulation (V^-, E^-, T^-). One can verify mechanically that $(V^+ \cup V^-, E^+ \cup E^-, T^+ \cup T^-)$ is a frugal triangulation of P.

1.5 Computing a Triangulation

The proof of the diagonal lemma implies an efficient algorithm to triangulate any simple polygon. I'll only sketch the algorithm here; for further details, see your favorite computational geometry textbook. First, we construct a trapezoidal decomposition in $O(n \log n)$ time using a *sweepline* algorithm. Intuitively, we sweep a vertical line from left to right across the plane, maintaining its intersection with the polygon in a balanced binary search tree, and inserting a new vertical wall whenever the line touches a vertex. (In fact, we only visit the vertices in order from left to right.) Second, we insert diagonals inside every boring trapezoid, decomposing P into monotone mountains in $O(n)$ time. Finally, we can triangulate each monotone mountain in $O(n)$ time by cutting off convex vertices in order from left to right.

The overall running time is $O(n \log n)$; the running time is dominated by the time to construct the trapezoidal decomposition. Theoretically faster algorithms for that construction are known—in particular, Chazelle described a famously complex $O(n)$_time algorithm—but it is unclear whether any of these improvements is faster in practice, or indeed if any of them have actually been implemented.

I'll leave the following corollaries of the polygon triangulation theorem as exercises.

Corollary: *Every frugal triangulation of a simple n-gon contains exactly $n-2$ triangles and exactly $n-3$ diagonals.*

Corollary: *Every simple polygon with at least four vertices has at least two* ***ears****, where an ear is an internal diagonal that cuts off a single triangle.*

Corollary: *Let P be a simple polygon with vertices $p_0, p_1, \ldots, p_{n-1}$. Let i, j, k, l be four distinct indices with $i < j$ and $k < l$, such that both $p_i p_j$ and $p_k p_l$ are interior diagonals of P. These two diagonals cross if and only if either $i < k < j < l$ or $k < i < l < j$.*

Corollary: *Any maximal set of non-crossing interior diagonals in a simple polygon P yields a frugal triangulation of P.*

1.6 The Dehn-Schönflies Theorem

The Dehn-Schönflies Theorem: *For any simple polygon P, there is a homeomorphism $H : \mathbb{R}^2 \to \mathbb{R}^2$ that maps P to a convex polygon Q and maps the interior of P to the interior of Q.*

Proof (Dehn): [to be written]

1.7 ...and the aptly named Sir Not Appearing in This Film

- Basic geometric algorithms:
  - Details of sweepline algortihm
  - Der Dreigroschenaltgorithmus
  - Faster decomposition/triangulation algorithms
- Triangulating polygons with holes
- Compatible triangulations
- Weakly simple polygons
- Proof (via Hex and Y) of the full Jordan Curve Theorem
- Geodesic polygons on other surfaces (see exercises)