> *Oh! marvellous, O stupendous Necessity–by thy laws thou dost compel every effect to be the direct result of its cause, by the shortest path. These [indeed] are miracles...*
>
> — Leonardo da Vinci, *Codex Atlanticus* (c. 1500)
> translated by Jean Paul Richter (1883)

> *Those who cannot remember the past are condemned to repeat it.*
>
> — George Santayana, *Reason in Common Sense* (1905)

> *A straight line may be the shortest distance between two points,*
> *but it is by no means the most interesting.*
>
> — The Doctor [Jon Pertwee], *The Time Warrior* (1973)

2 Shortest Homotopic Paths

2.1 A Few Definitions

Let X be any topological space. A ***path*** in X is a continuous function from the unit interval $[0,1]$ to X, and a ***cycle*** in X is a continuous function from the standard unit circle $\boldsymbol{S^1} := \{(x,y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\}$ to X. A ***loop*** is a path whose endpoints coincide; this common endpoint is called the ***basepoint*** of the loop. A path or cycle is ***simple*** if it is injective; a loop is simple if its restriction to $[0,1)$ is injective. We refer to paths, cycles, and loops collectively as ***curves***.

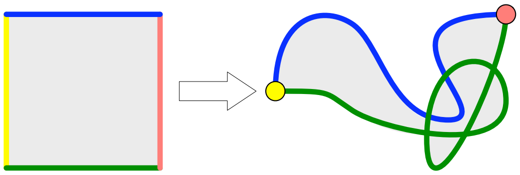
The ***Jordan-Schönflies Theorem*** states that for any simple cycle γ in the plane, there is a homeomorphism from the plane to itself whose restriction to S^1 is γ. Thus, the image of any simple cycle partitions the plane into two components, a bounded ***interior*** whose closure is homeomorphic to the disk B^2, and an unbounded ***exterior***.

For any paths π and π' with $\pi(1) = \pi'(0)$, the ***concatenation*** $\boldsymbol{\pi \cdot \pi'}$ is the path

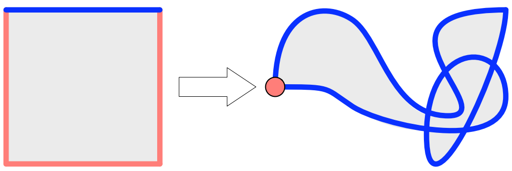
$$(\pi \cdot \pi')(t) := \begin{cases} \pi(2t) & \text{if } t \le 1/2, \\ \pi'(2t-1) & \text{if } t \ge 1/2. \end{cases}$$

The ***reversal*** $\boldsymbol{\overline{\pi}}$ of a path π is the path $\overline{\pi}(t) := \pi(1-t)$.

A ***path homotopy*** between paths π and π' is a continuous function $h\colon [0,1] \times [0,1] \to X$ such that $h(0,t) = \pi(t)$ and $h(1,t) = \pi'(t)$ for all t, and $h(s,0) = \pi(0) = \pi'(0)$ and $h(s,1) = \pi(1) = \pi'(1)$ for all $s \in [0,1]$. (We will omit the word 'path' when it is clear from context.) For all $t \in [0,1]$, the function $s \mapsto h(s,t)$ is a path from $\pi(0)$ to $\pi(1)$. Two paths π and π' are ***(path) homotopic*** if there is a path homotopy between them; we write $\boldsymbol{\pi \simeq \pi'}$ to denote that π and π' are homotopic. Tedious definition-chasing implies that $\simeq$ is an equivalence relation. We refer to the equivalence classes as ***homotopy classes***, and write $[\pi]$ to denote the homotopy class of path π.



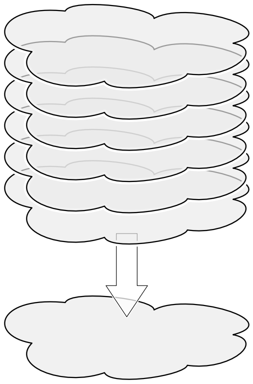
A homotopy between two paths.



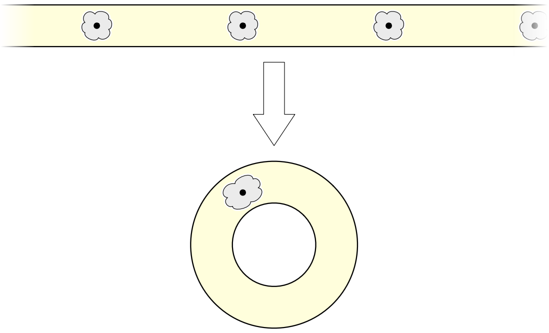
A homotopy from a contractible loop to its basepoint.

We call a loop ℓ is ***contractible*** if it is path-homotopic to the constant path mapping the entire interval $[0,1]$ to the basepoint $\ell(0)$. Two paths π and π' with the same endpoints are homotopic if and only if the loop $\pi \cdot \overline{\pi}'$ is contractible. A connected topological space X is ***simply connected*** if every loop in X is contractible. For example, the plane and the sphere are both simply connected, but the annulus and the torus are not.

A ***covering map*** is a continuous surjection $p\colon \hat{X} \to X$ such that any point $x \in X$ has an open neighborhood U whose preimage $p^{-1}(U)$ can be written as the union of disjoint open sets $\bigsqcup_{i\in I} U_i$, and the restriction of p to each set U_i is a homeomorphism to U. If there is a covering map from a space $\hat{X}$ to another space X, we call $\hat{X}$ a ***covering space*** of X. As a trivial example, X is a covering space of X, with the identity function (or any homeomorphism) as the covering map. We implicitly consider only *connected* covering spaces, to avoid trivial cases like the disjoint union of several copies of X.



The local behavior of every covering map.



The infinite strip is the universal cover of the annulus.

The ***universal covering space*** $\tilde{X}$ of X is the unique simply-connected covering space of X. If $\hat{X}$ is a connected covering space of X, then $\tilde{X}$ is also a (universal) covering space of $\hat{X}$. The universal covering space $\tilde{X}$ can be described as the set of all homotopy classes of paths starting at an arbitrary fixed ***basepoint*** $x \in X$:

$$\tilde{X} := \{[\pi] \mid \pi\colon [0,1] \to X \text{ and } \pi(0) = x\}.$$

(The choice of basepoint is not important; different basepoints lead to homeomorphic covering spaces.) The associated covering map $p\colon \tilde{X} \to X$ maps any homotopy class to its final endpoint: $p([\pi]) = \pi(1)$. For example, the plane is its own universal covering space, as is the sphere. The universal cover of the closed annulus $\{(x, y) \mid 1 \le x^2 + y^2 \le 2\}$ is homeomorphic the infinite strip $\{(x, y) \mid 1 \le x \le 2\}$.

2.2 Shortest (Homotopic) Paths in Polygons

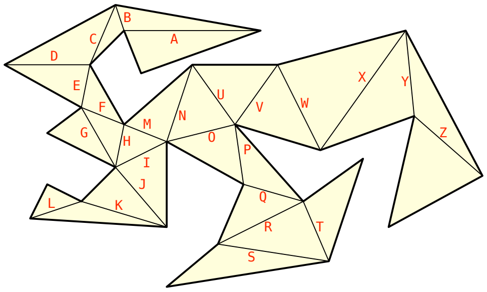
The ***shortest homotopic path problem*** can be described as follows. The input consists of a topological *metric* space X, along with a path π in X, and the desired output is a path $\bar{\pi}$ of minimum length that is homotopic to π.

In this lecture we consider an algorithm for a concrete special case of this problem, originally due to Hershberger and Snoeyink [6]. We call a curve in the plane ***polygonal*** if its image is the union of a finite number of line segments. A ***(simple) polygon*** is the closure of the interior of a simple polygonal closed curve; when we want to emphasize the curve itself, we refer to the ***boundary*** of the polygon. The input to our algorithm is a simple polygon P and a polygonal path π (which may not be simple). Let n denote the number of edges of P, and let k denote the number of segments in π.

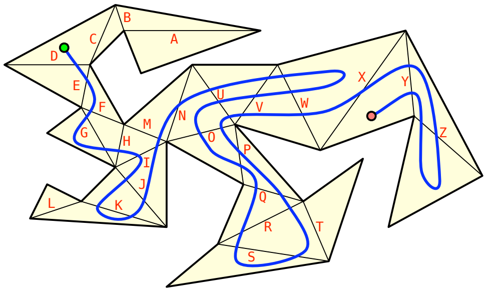
The Jordan-Schönflies theorem implies that P is homeomorphic to a disk; it follows easily that P is simply connected. Thus, the shortest path $\bar{\pi}$ homotopic to π is just the shortest path in P from $\pi(0)$ to $\pi(1)$. Nevertheless, we will approach this special case as though its topology were nontrivial, because it illustrates important concepts that are useful in more general settings.

Without loss of generality, we will assume that the input polygon P has been triangulated. A ***triangulation*** of P is a decomposition of P into a finite set of triangles, such that every triangle vertex is a vertex of P, and the intersection of any two triangles is either an edge of both, a vertex of both, or the empty set. (A polygon triangulation is a simple example of a *simplicial complex*.)

Any simple polygon with n edges can be triangulated in $O(n)$ time [1, 3]; however, the hidden constants are large, and the algorithms are so complicated that they have no hope of being implemented. There are much simpler randomized algorithms with expected running time $O(n \log^* n)$ that are almost certainly more efficient in practice [4, 5, 8].



A labeled polygon triangulation



A path with crossing sequence
DEFGGHIJKKJIMNUVWWVUOPQRSSRQPOVWXYZZY

2.2.1 Crossing Sequences

The first step of the algorithm is to compute the ***crossing sequence*** of the input path π, which we denote $X(\pi)$. This is the sequence of diagonals that a point moving continuously along π crosses; we associate a unique label with each diagonal in the input triangulation. (We define 'cross' exactly as in the point-in-polygon algorithm; in particular, we need to be careful when a vertex of π lies directly on a diagonal of the triangulation.) Let x denote the number of symbols in the crossing sequence. Each segment of π crosses each diagonal in the triangulation at most once, so $x = O(nk)$.

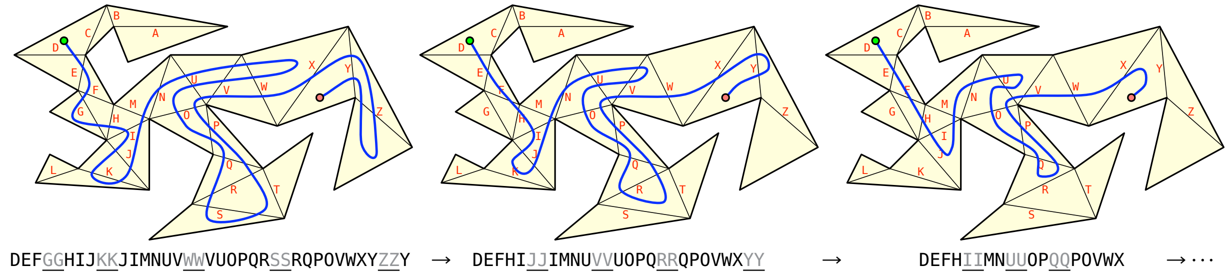
2.2.2 Reduction

Next the algorithm ***reduces*** the crossing sequence by repeatedly removing any adjacent pairs of the same label. Intuitively, reducing the crossing sequence mirrors a homotopy of the input path that reduces the length of the path, and therefore the number of edge crossings. For example, we can reduce the crossing sequence in the figure above as follows:

$$\texttt{DEF}\underline{\texttt{GG}}\texttt{HIJ}\underline{\texttt{KK}}\texttt{JIMNUV}\underline{\texttt{WW}}\texttt{VUOPQR}\underline{\texttt{SS}}\texttt{RQPOVWXY}\underline{\texttt{ZZ}}\texttt{Y}$$
$$\longrightarrow \texttt{DEFHI}\underline{\texttt{JJ}}\texttt{IMNU}\underline{\texttt{VV}}\texttt{UOPQ}\underline{\texttt{RR}}\texttt{QPOVWX}\underline{\texttt{YY}}$$
$$\longrightarrow \texttt{DEFH}\underline{\texttt{II}}\texttt{MN}\underline{\texttt{UU}}\texttt{OP}\underline{\texttt{QQ}}\texttt{POVWX}$$
$$\longrightarrow \texttt{DEFHMNO}\underline{\texttt{PP}}\texttt{OVWX}$$
$$\longrightarrow \texttt{DEFHMN}\underline{\texttt{OO}}\texttt{VWX}$$
$$\longrightarrow \texttt{DEFHMNVWX}$$

The reduction step is more formally justified by the following lemma. Recall that $\bar{\pi}$ is the path we are trying to compute: the shortest path homotopic to π.

Lemma 2.1. *Reducing the crossing word $X(\pi)$ yields the crossing word $X(\bar{\pi})$.*


DEFGGHIJKKJIMNUVWWVUOPQRSSRQPOVWXYZZY →
DEFHIJJIMNUVVUOPQRRQPOVWXYY →
DEFHIIMNUUOPQQPOVWX →···

Proof: Call a string *reduced* if it contains no repeating pairs, and call two strings *equivalent* if we can transform one into the other by a sequence of insertions and deletions of repeating pairs. The lemma follows from three observations:

First, any two homotopic paths have equivalent crossing sequences. As we continuously deform one path to the other, the crossing sequence changes only at certain discrete critical values and only by the insertion or deletion of repeating pairs. It is absolutely crucial here that the vertices of the triangulation all lie on the boundary of P.

Second, every string is equivalent to exactly one reduced string, called its ***reduction***. An easy induction argument implies that any two equivalent strings have the same reduction. Thus, any two homotopic paths have the same reduced crossing word.

Finally, if π crosses any edge e twice in a row, we can make π shorter by homotoping the subpath between the two crossing points onto e. We conclude that the crossing sequence of $\bar{\pi}$ is reduced. □

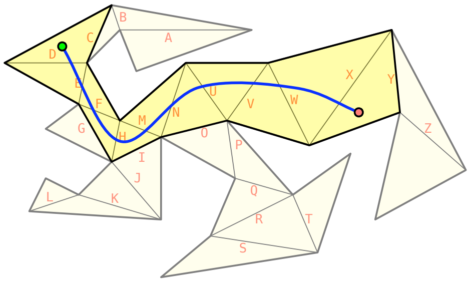
The crossing word can be reduced in $O(x)$ time using the following algorithm. Here, • is a special sentinel symbol that is different from every edge label.

```
REDUCE($X[1..x]$):
  $\bar{x} \leftarrow 0$
  $\bar{X}[0] \leftarrow \bullet$
  for $i \leftarrow 1$ to $x$
      if $X[i] = \bar{X}[\bar{x}]$
          $\bar{x} \leftarrow \bar{x} - 1$          ⟨⟨pop⟩⟩
      else
          $\bar{x} \leftarrow \bar{x} + 1$
          $\bar{X}[\bar{x}] \leftarrow X[i]$        ⟨⟨push⟩⟩
  return $\bar{X}[1..\bar{x}]$
```

We can also characterize the reduced crossing sequence in terms of parity. For any diagonal edge e, the Jordan Curve Theorem implies that $P \setminus e$ has exactly two components. If an edge label e occurs an odd number of times in $X(\pi)$, then $\pi(0)$ and $\pi(1)$ lie in different components of $P \setminus e$; thus *any* path from $\pi(0)$ to $\pi(1)$ must cross e, and the shortest path must cross e exactly once. On the other hand, if an edge label e occurs an even number of times in $X(\pi)$, then $\pi(0)$ to $\pi(1)$ are in the same component of $P \setminus e$, so that shortest path $\bar{\pi}$ does not cross e at all. Thus, the reduced crossing sequence $\bar{X}$ contains precisely the edge labels that appear an odd number of times in $X(\pi)$; moreover, these labels are sorted by their first (or last) occurrence in $X(\pi)$.

2.2.3 Sleeve

Let $\bar{X}$ denote the reduced crossing sequence, and let $\bar{x}$ denote its length. With $\bar{X}$ in hand, we can restrict our attention to a subset of the triangles. The crossing sequence defines a sequence of $\bar{x} + 1$ triangles, starting with the triangle containing $\pi(0)$ and ending with the triangle containing $\pi(1)$. The ***sleeve*** of $\bar{X}$ is constructed by gluing together copies of the triangles in this sequence along their common edges.

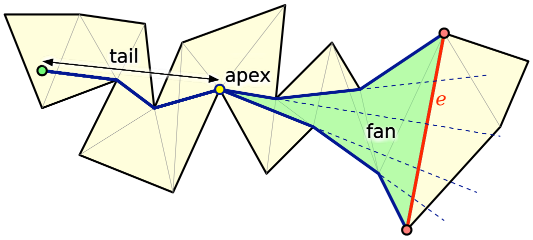


The sleeve for the reduced crossing word DEFHMNVWX.

If $\bar{x} = 0$, the sleeve consists of a single triangle, and the shortest path from $\pi(0)$ to $\pi(1)$ is a simple line segment. So assume from now on that $\bar{x} > 0$. We can clearly construct the sleeve in $O(\bar{x})$ time. Lemma 2.1 implies that $\bar{\pi}$ is the shortest path within the sleeve from $\pi(0)$ to $\pi(1)$.

2.2.4 Funnel

Finally, we compute this shortest path using an algorithm independently proposed by Chazelle [2] and by Lee and Preparata [7]. The ***funnel*** of any diagonal e of the sleeve is the union of shortest paths from $\pi(0)$ to all points on e. The funnel consists of a polygonal path, called the ***tail***, from $\pi(0)$ to a point a called the ***apex***, plus a simple polygon called the ***fan***. The tail may be empty, in which case $\pi(0)$ is the apex. The fan is bounded by the edge e and two concave chains joining the apex to the endpoints of e. The shortest path from $\pi(0)$ to either endpoint of e consists of the tail plus one of the concave chains bounding the fan. Extending the edges of the concave chains to infinite rays defines a series of ***wedges***, which subdivide not only the fan but the triangle just beyond e.

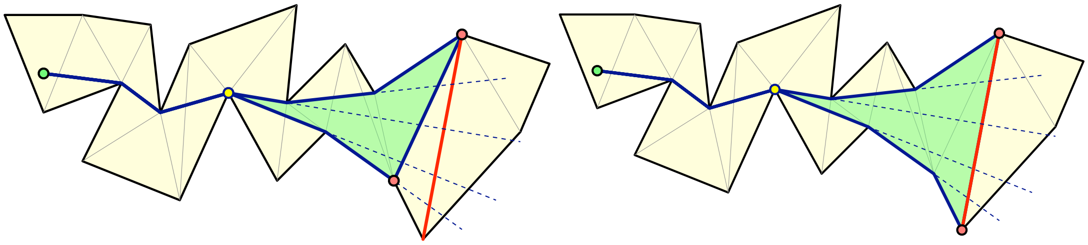


Anatomy of a typical funnel.

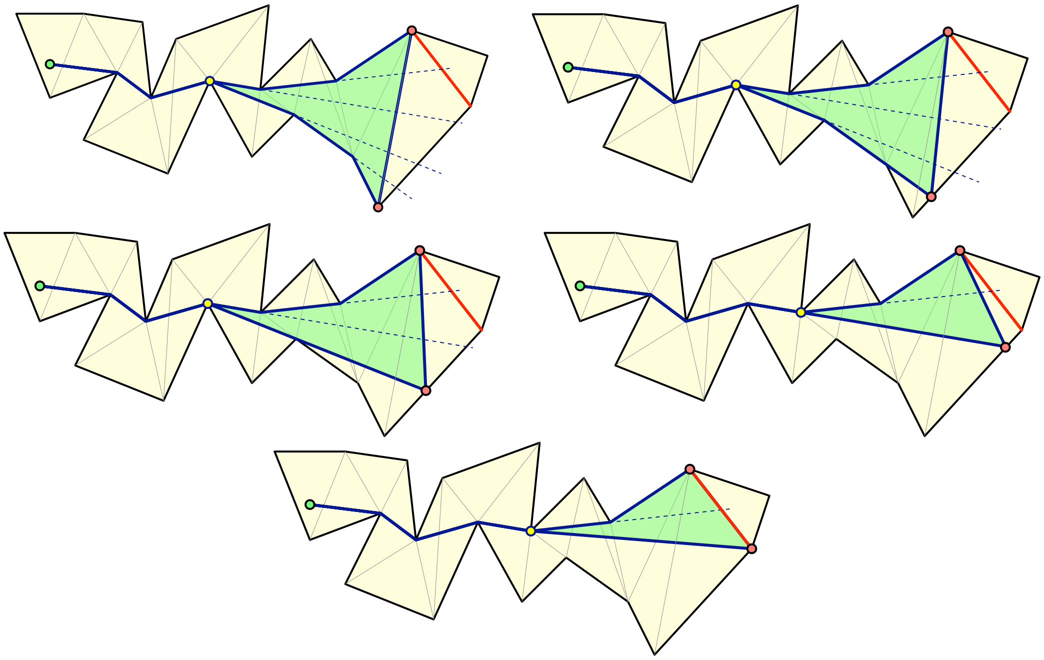
Beginning with a single triangle joining $\pi(0)$ to the first edge in $\bar{X}$, we extend the funnel through the entire sleeve one diagonal edge at a time. Each diagonal edge shares one endpoint with the previous edge; suppose we are extending the funnel from edge uv to edge vw. There are two cases to consider.

Let t be the predecessor of u on the shortest path from $\pi(0)$ to u. If the points v and w lie on opposite sides of the ray $\overrightarrow{ut}$, then the new endpoint w does not lie inside any wedge of the current fan. We can detect this case in $O(1)$ time with a single orientation test, and then *extend* the tunnel in $O(1)$ time by inserting w as a new fan vertex.

Otherwise, we *contract* the funnel, intuitively by moving u continuously along the boundary edge uw. Each time the point crosses the boundary of a wedge, we remove a vertex from the fan. If the removed vertex is the apex, its successor on the shortest path from $\pi(0)$ to v becomes the new apex, and the tail grows by an edge. We can detect whether the moving point will cross any wedge boundary in $O(1)$ time using our standard orientation test. Thus, the total time in this case is $O(d+1)$, where d is the number of vertices deleted from the fan. However, the total number of *deleted* vertices cannot exceed the total number of *previously inserted* vertices, so the *amortized* time to process this case is also $O(1)$.



Extending and widening the funnel.



Extending and narrowing the funnel; the apex moves in the fourth step.

2.2.5 Conclusion

When the funnel has reached the last edge in $\bar{X}$, we compute the shortest path from $\pi(0)$ to $\pi(1)$ in the sleeve by treating $\pi(1)$ as a triangle vertex and extending the funnel one last time. Thus, assuming the polygon is already triangulated, the overall time to compute the shortest path is $O(x + \bar{x}) = O(x) = O(nk)$.

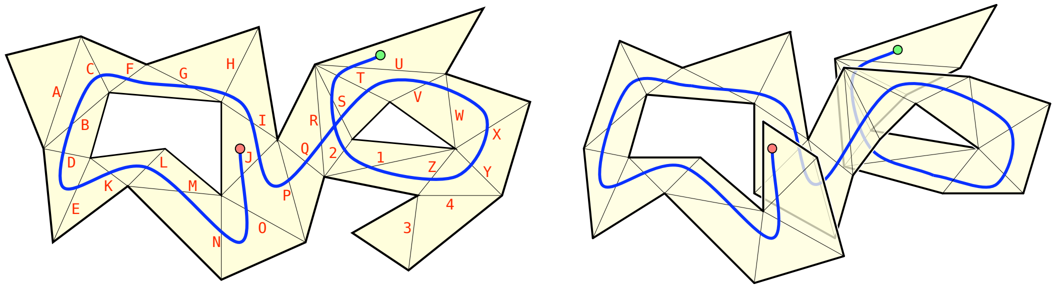
In an actual implementation, it is not necessary to separate the algorithm into separate crossing sequence, reduction, sleeve, and funnel phases. Instead, it is possible to compute the tree of shortest paths from $\pi(0)$ to every vertex of every triangle crossed by the input path π in $O(x)$ time using single transversal of π, after which the shortest homotopic path to $\pi(1)$ can be extracted in $O(\bar{x})$ time.

2.3 Shortest Homotopic Paths in Polygons with Holes

Hershberger and Snoeyink [6] actually solve the shortest homotopic path problem for a much more general class of spaces than simple polygons. A ***polygon with holes*** is a connected planar region whose boundary consists of two or more disjoint simple polygonal closed curves. One of these curves is the *outer boundary* of P. The other curves all lie in the interior of the outer boundary and have disjoint interiors, which are called the *holes* of P. Unlike simple polygons, polygons with holes are *never* simply connected; in particular, the boundary of any hole is non-contractible.

Surprisingly, the shortest homotopic path algorithm for simple polygons can be applied to polygons with holes with (almost) no modifications! As before, let n denote the number of edges in P, and let K denote the number of segments in π.

- As in the previous section, we assume that we are given a triangulation of P, with all triangle vertices on the boundary of P. Otherwise, we can triangulate P in $O(n \log n)$ time, or even in $O(n \log^* n + h \log n)$ expected time, where h is the number of holes [8]. This is the only change in the algorithm or its running time.
- We can still compute the crossing sequence $X(\pi)$ in $O(x)$ time, where $x = O(kn)$ is the number of crossings.
- We can still reduce the crossing sequence in $O(x)$ time, using the same algorithm. The proof of Lemma 2.1 *never* used the fact that P is simply-connected, so the lemma applies without modification to polygons with holes. We emphasize that the reduced crossing sequence $\bar{X}$ can still contain the same edge label more than once, just not twice in succession.
- We can still compute the sleeve of $\bar{X}$ in $O(\bar{x})$ time, where $\bar{x}$ is the length of $\bar{X}$. Each time a path following $\bar{X}$ enters a triangle, we add ***a new copy*** of that triangle to the evolving sleeve. Thus, if a reduced path enters the same triangle five times, the resulting sleeve contains five different copies of that triangle. The sleeve is no longer a triangulated *simple* polygon; however, it is still homeomorphic to a disk. Moreover, if we represent the sleeve as a linked list of triangles, any self-overlaps are simply irrelevant.



A reduced path in a polygon with two holes, with and the resulting non-simple sleeve.
The crossing sequence of the path is `UTS21ZYWVTSRQPJIHGFCBDEKLMNOJ`.

- We can still compute the shortest path in the sleeve using the funnel algorithm in $O(\bar{x})$ time. Even in this more general setting, the *fan* is always a simple polygon, and so each extension step can be carried out exactly as described. The tail may intersect itself or the fan any number of times, but the algorithm won't notice.
- The running time of the algorithm, assuming the input space is triangulated, is still $O(x + \bar{x}) = O(x) = O(nk)$.

Another useful way to think about the behavior of the algorithm is that it cannot distinguish between the original polygon with holes P and its universal cover $\tilde{P}$. The correspondence between homotopy classes of paths and reduced crossing sequences implies the following description of $\tilde{P}$ in terms of the triangulation of P. We describe an infinite triangulation of $\tilde{P}$ by listing its constituent triangles and then declaring which pairs of edges should be identified.

Fix an arbitrary *basepoint* $p \in P$. Call a string X of edge labels *legal* if it is the reduced crossing sequence of a path π_X with $\pi_X(0) = p$. For each legal string X, let Δ_X denote ***a unique copy of*** the triangle containing $\pi_X(1)$. Each triangle Δ lifts to an infinite number of triangles Δ_X, one for each reduced crossing sequence ending in Δ. If X and Y are legal strings with $Y = Xe$, the triangles Δ_X and Δ_Y each contain a copy of edge e on their boundary; call these copies e_X and e_Y.

> The universal cover $\tilde{P}$ is obtained from the disjoint union of all triangles Δ_X by identifying all pairs of edges e_X and e_Y such that $X = Ye$ for some edge e.

The choice of basepoint is unimportant; different basepoints induce different legal crossing sequences and therefore differently labeled triangles, but the resulting infinite triangulations are isomorphic.

The resulting triangulation of $\tilde{P}$ is infinite, but this is not a problem—our shortest (homotopic) path algorithm only examines the *finite* set of triangles that intersect the input path π.

We can also describe the transformation from P to $\tilde{P}$ strictly in terms of crossing sequences. In any crossing sequence X, call two edge labels $X[i]$ and $X[j]$ *equivalent* if the substring $X[i\,..\,j]$ is equivalent to the empty string. For example, in the crossing sequence ABCCCBCABCCBACBBC, we can indicate equivalent labels with subscripts: $\mathtt{A_1B_1C_1C_1C_1B_2C_2A_2B_3C_3C_3B_3A_2C_2B_2B_2C_2}$. Then the reduced crossing sequence is simply the subsequence of distinct labels that occur an odd number of times, in order by their first occurrence, exactly as in the simply polygon case. For example, the crossing sequence $\mathtt{A_1B_1C_1C_1C_1B_2C_2A_2B_3C_3C_3B_3A_2C_2B_2B_2C_2}$ reduces to $\mathtt{A_1B_1C_1B_2C_2} = \mathtt{ABCBC}$. Nonequivalent labels that refer to the same edge 'really' refer to two different lifts of that edge in the universal cover.

References

[1] N. M. Amato, M. T. Goodrich, and E. Ramos. A randomized algorithm for triangulating a simple polygon in linear time. *Discrete Comput. Geom.* 26:245–265, 2001.

[2] B. Chazelle. A theorem on polygon cutting with applications. *Proc. 23rd Annu. IEEE Sympos. Found. Comput. Sci.*, 339–349, 1982.

[3] B. Chazelle. Triangulating a simple polygon in linear time. *Discrete Comput. Geom.* 6(5):485–524, 1991.

[4] K. L. Clarkson, R. E. Tarjan, and C. J. V. Wyk. A fast Las Vegas algorithm for triangulating a simple polygon. *Discrete Comput. Geom.* 4:423–432, 1989.

[5] O. Devillers. Randomization yields simple $O(n \log^* n)$ algorithms for difficult $\Omega(n)$ problems. *Int. J. Comput. Geom. Appl.* 2(1):621–635, 1992.

[6] J. Hershberger and J. Snoeyink. Computing minimum length paths of a given homotopy class. *Comput. Geom. Theory Appl.* 4:63–98, 1994.

[7] D.-T. Lee and F. P. Preparata. Euclidean shortest paths in the presence of rectilinear barriers. *Networks* 14:393–410, 1984.

[8] R. Seidel. A simple and fast incremental randomized algorithm for computing trapezoidal decompositions and for triangulating polygons. *Comput. Geom. Theory Appl.* 1(1):51–64, 1991.