

COMPUTATIONAL GEOMETRY

Triangulating Monotone Mountains

Definitions and Properties of Monotonicity

We say that a simple polygon P is *monotone* in direction d (a vector, or an oriented line), or *d -montone*, if any line orthogonal to d intersects P in a connected set (i.e., a line segment, a point, or the empty set). We say that P is *strictly monotone* in direction d (or *strictly d -monotone*) if P is monotone in direction d and, for any line ℓ orthogonal to d , if $\ell \cap P \neq \emptyset$, then the endpoints of the line segment $ab = \ell \cap P$ clearly see one another (i.e., $ab \cap \partial P \subseteq \{a, b\}$). (Note that we allow $a = b$.)

If P is strictly monotone in direction d , then there is a unique vertex, b (the “bottom”), with smallest d -coordinate (obtained by projecting points of P orthogonally onto the oriented line through d); similarly, there is a unique vertex, t (the “top”), with largest d -coordinate. If P is d -monotone (but not necessarily strictly monotone), then there will be at most two vertices of P with smallest (resp., largest) d -coordinate; we can take b (resp., t) to be either one of them, if there is a tie. If P is d -monotone, the points b and t partition the boundary, ∂P , into two d -monotone (or strictly d -monotone) polygonal chains, often called the *left chain* (going counter-clockwise from b to t) and the *right chain* (going clockwise from b to t).

We say that P is a *d -monotone mountain* if P is d -monotone and either the left chain or the right chain is a single line segment (for some choice of b and t , if they are not uniquely determined). A similar definition applies to the case of a *strictly d -monotone mountain*. The segment $bt \subset \partial P$ is called the *base* of the d -monotone mountain P .

A simple polygon P is called *monotone* if there exists a direction d for which it is d -monotone. Similarly, P is *strictly monotone* if there exists a direction d for which P is strictly d -monotone, and P is a *monotone mountain* (or a *strictly monotone mountain*) if there is a direction d for which it is a d -monotone mountain (or strictly d -monotone mountain). In general, the set of directions d for which a polygon is monotone can be associated with arcs of the unit circle centered on the origin (i.e., with angles between 0 and 360). We let $M(P)$ denote the set of unit vectors (points on the unit circle centered at the origin) with respect to which P is d -monotone. In the case of strict monotonicity, the arcs will be *open* (not include their endpoints); for just monotonicity, the arcs are closed (they include their endpoints). Note that, from the definition, if P is d -monotone, then it is necessarily $(-d)$ -monotone; thus, the arcs of monotonicity for P come in pairs.

It is easy to give examples of polygons P for which there are multiple arcs of monotonicity; in fact, $M(P)$ can have a linear number of components ($\Omega(n)$) in some cases. It is not hard to see that $M(P)$ has $O(n)$ components (a linear upper bound), since the endpoints of the arcs of monotonicity correspond to directions orthogonal to edges of P .

Lemma 1. A simple polygon P is convex if and only if it is d -monotone for any direction d (i.e., $M(P) =$ the unit circle).

For a simple polygon P , we define an *interior cusp* with respect to direction d to be a (strictly) reflex vertex, v_i , of P such that v_{i-1} and v_{i+1} are both at or below or both at or above v_i in d -coordinate.

Lemma 2 [Lemma 2.1.1, O’Rourke]. If simple polygon P has no interior cusp with respect to direction d , then P is d -monotone.

We say that polygon P is *y -monotone* (or strictly *y -monotone*, or a *y -monotone mountain*, etc) if P is d -monotone with respect to direction $d = (0, 1)$ (i.e., with respect to the (oriented) y -axis). A similar definition applies to x -monotonicity.

Triangulating Monotone Simple Polygons

It is known that any simple polygon P can be triangulated in linear time $O(n)$ using the highly complex algorithm of Chazelle. Simple algorithms to triangulate a simple polygon in linear time are known for several special classes of simple polygons, including: convex polygons, star-shaped (i.e., 1-guardable) polygons, monotone polygons, and others. In fact, it is trivial to triangulate a convex polygon in linear ($O(n)$) time: a “fan triangulation” works, from any vertex v_i .

Here, we describe a very simple algorithm to triangulate monotone mountains (a special case of monotone polygons). The algorithm relies on the following simple fact about the very special structure of monotone mountains:

Lemma 3 [Lemma 2.3.1, O'Rourke]. Any strictly convex vertex v_i of a monotone mountain, other than possibly b or t , is an ear tip; i.e., if $v_i \notin \{b, t\}$ is strictly convex, then $v_{i-1}v_{i+1}$ is a diagonal (v_{i-1} clearly sees v_{i+1}).

The algorithm simply clips off ears one by one. Since it now takes time $O(1)$ to test if v_i is an ear tip (instead of $O(n)$, using $\text{Diagonal}(v_{i-1}, v_{i+1})$), the algorithm runs in time $O(n)$ ($O(1)$ per ear tip cut off). When we cut off an ear, we have only to check/update the ear tip status of the two endpoints of the ear diagonal (this costs $O(1)$). While any order of processing candidate ear tips works fine, to make it explicit, so that every one of us gets the exact same set of diagonals in the same order, we specify the algorithm precisely in such a way that we always obey the following rule: Among all non-base-endpoint vertices that are candidate ear tips (i.e., that are strictly convex), chose the one with highest y -coordinate to cut off next.

Let P be a y -monotone mountain with base bt . Relabel the vertices so that $u_0 = t, u_1, u_2, \dots, u_{n-1} = b$ are the vertices given in *decreasing* order of y -coordinate.

Algorithm: Triangulate-Monotone-Mountain()

Step 0 $i \leftarrow 1$

Step 1 If $i = n - 2$, exit. If u_i is strictly convex, then draw diagonal $u_{i-1}u_{i+1}$ to cut off the ear tip u_i . If it is now the case that u_{i-1} is strictly convex, and $u_{i-1} \neq t$, then $i \leftarrow i - 1$; otherwise, $i \leftarrow i + 1$. Go to Step 1.

Example For the monotone mountain shown below, show the (unique) triangulation that is given by the algorithm *Triangulate-Monotone-Mountain()*.

We indicate the order in which diagonals were inserted by writing an index next to each diagonal.

