14. Single faces in arrangements

Lecture on Monday 17$^{\text{th}}$ November, 2008 by Michael Hoffmann <hoffmann@inf.ethz.ch>

14.1 Constructing lower envelopes

Theorem 14.1 *Let $\mathcal{F} = \{f_1, \ldots, f_n\}$ be a collection of real-valued continuous functions defined on a common interval $I \subset \mathbb{R}$ such that no two functions from $\mathcal{F}$ intersect in more than s points. Then the lower envelope $\mathcal{L}_\mathcal{F}$ can be constructed in $O(\lambda_s(n) \log n)$ time. (Assuming that an intersection between any two functions can be constructed in constant time.)*

Proof. Divide and conquer. For simplicity, assume that n is a power of two. Split $\mathcal{F}$ into two almost equal parts $\mathcal{F}_1$ and $\mathcal{F}_2$ and construct $\mathcal{L}_{\mathcal{F}_1}$ and $\mathcal{L}_{\mathcal{F}_2}$ recursively. The resulting envelopes can be merged using line sweep by processing $2\lambda_s(n/2)+\lambda_s(n) \leq 3\lambda_s(n)$ events. (The first term accounts for events generated by the vertices of the two envelopes to be merged. The second term accounts for their intersections, each of which generates a vertex of the resulting envelope.) Observe that no sorting is required and the SLS structure is of constant size. Therefore, the sweep can be done in time linear in the number of events.

This yields the following recursion for the runtime $T(n)$ of the algorithm. $T(n) \leq 2T(n/2) + c\lambda_s(n)$, for some constant $c \in \mathbb{N}$. Observe that $k\lambda_s(n/k) \leq \lambda_s(n)$, for $k | n$, because any k DS-sequences on an alphabet of size n/k can be concatenated to a single DS-sequence on an alphabet of size n by using pairwise disjoint (parts of the) alphabets for each of the k sequences. It follows that $T(n) \leq c \sum_{i=1}^{\log n} 2^i \lambda_s(n/2^i) = c \sum_{i=1}^{\log n} \lambda_s(n) = O(\lambda_s(n) \log n)$. $\square$

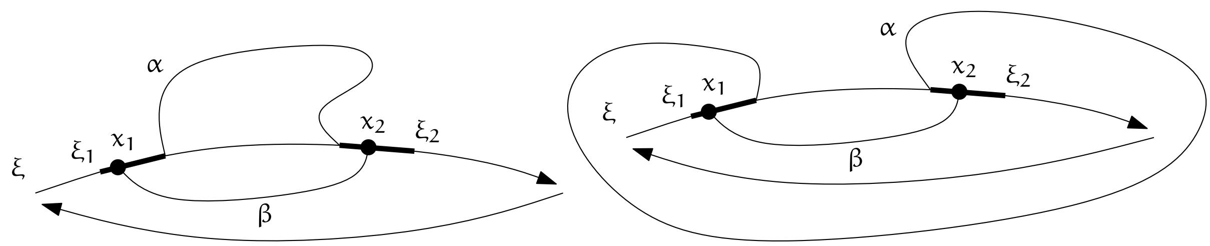
14.2 Complexity of a single face

Theorem 14.2 *Let $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$ be a collection of Jordan arcs in $\mathbb{R}^2$ such that each pair intersects in at most s points, for some $s \in \mathbb{N}$. Then the combinatorial complexity of any single face in the arrangement $\mathcal{A}(\Gamma)$ is $O(\lambda_{s+2}(n))$.*

Proof. Consider a face f of $\mathcal{A}(\Gamma)$. In general, the boundary of f might consist of several connected components. But as any single curve can appear in at most one component we may as well suppose that the boundary consists of one component only. (The complexity we are heading for is super-linear.)

Replace each γ_i by two directed arcs γ_i^+ and γ_i^- that together form a closed curve that is infinitesimally close to γ_i. Denote by S the circular sequence of these oriented curves, in their order along the (oriented) boundary ∂f of f.

Consistency Lemma. Let ξ be one of the oriented arcs γ_i^+ or γ_i^-. The order of portions of ξ that appear in S is consistent with their order along ξ. (That is, for each ξ we can break up the circular sequence S into a linear sequence $S(\xi)$ such that the



(a) f lies on the unbounded side of $\alpha \cup \beta$. (b) f lies on the bounded side of $\alpha \cup \beta$.

Figure 14.1: *Cases in the Consistency Lemma.*

portions of ξ that correspond to appearances of ξ in $S(\xi)$ appear in the same order along ξ.)

Consider two portions ξ_1 and ξ_2 of ξ that appear consecutively in S. Choose points $x_1 \in \xi_1$ and $x_2 \in \xi_2$ and connect them in two ways: first by the arc α following ∂f as in S, and second by an arc β inside the closed curve formed by γ_i^+ or γ_i^-. The curves α and β do not intersect except at their endpoints and they are both contained in the complement of the interior of f. In other words, $\alpha \cup \beta$ forms a closed Jordan curve and f lies either in the interior of this curve or in its exterior. In either case, the part of ξ between ξ_1 and ξ_2 is separated from both x and y by $\alpha \cup \beta$. Therefore, ξ_1 and ξ_2 are also consecutive boundary parts in the order of boundary portions along ξ, which proves the lemma.

Break up S into a linear sequence $S' = (s_1, \ldots, s_t)$ arbitrarily. For each oriented arc ξ, consider the sequence $s(\xi)$ of its portions along ∂f in the order in which they appear along ξ. By the Consistency Lemma, $s(\xi)$ corresponds to a subsequence of S, starting at s_k, for some $1 \leq k \leq t$. In order to consider $s(\xi)$ as a subsequence of S', break up the symbol for ξ into two symbols ξ and ξ' and replace all occurrences of ξ in S' before s_k by ξ'. Doing so for all oriented arcs results in a sequence S^* on at most $4n$ symbols.

Claim: S^* is a $(4n, s+2)$-Davenport-Schinzel sequence.

Clearly no two adjacent symbols in S^* are the same. Suppose S^* contains an alternating subsequence $\xi \ldots \eta \ldots \xi \ldots \eta$ of length $s+4$. Consider any four consecutive elements of this subsequence. Choose points $x, y \in \xi$ and $z, w \in \eta$ such that the appear in the order x, z, y, w along ∂f. Connect x and y by a Jordan arc β_{xy} within the closed curve formed by ξ and its counterpart. Similarly, connect z and w by a Jordan arc β_{zw} within the closed curve formed by η and its counterpart. Then connect x, z, y, w along ∂f by curves β_{xz}, β_{zy}, β_{yw}, and β_{wx}. Observe that the last four curves are pairwise disjoint except for common endpoints. Moreover, none of them intersects β_{xy} or β_{zw}, except at a common endpoint.

We claim that β_{xy} and β_{zw} intersect. Suppose they do not. Then the six curves $\beta_.$ form a plane graph on x, y, z, w which together with a point u chosen somewhere inside f and curves/edges connecting u to all of x, y, z, w within f form a plane embedding of K_5, contradiction.

In other words, any quadruple of consecutive elements from the alternating subsequence induces an intersection between the corresponding arcs ξ and η. Clearly these intersection points are pairwise distinct for any pair of distinct quadruples which altogether provides $s+4-3=s+1$ points of intersection between ξ and η, in contradiction to the assumption that they intersect in at most s points. □

Corollary 14.3 *The combinatorial complexity of a single face in an arrangement of* n *line segments in* $\mathbb{R}^2$ *is* $O(\lambda_3(n)) = O(n\alpha(n))$.

14.3 Constructing a single face

Theorem 14.4 *Given a set* S *of* n *line segments and a point* $x \in \mathbb{R}^2$*, the face of* $\mathcal{A}(S)$ *that contains* x *can be constructed in* $O(\lambda_3(n) \log n)$ *time.*

Phrased in terms of translational motion planning this means the following.

Corollary 14.5 *Consider a simple polygon* R *with* k *edges (robot) and a polygonal environment* $\mathcal{P}$ *that consists of* n *edges in total. The free space of all positions of* R *that can be reached by translating it without intersecting an obstacle from* $\mathcal{P}$ *has complexity* $O(\lambda_3(kn))$ *and it can be constructed in* $O(\lambda_3(kn) \log(kn))$ *time.*

We prove Theorem 14.4 using a randomized incremental construction. In fact, we construct the trapezoidal map induced by the given set S of segments, which is defined in the same way as for disjoint segments. The only difference is that here we also subdivide using vertical rays emanating from intersection points of segments. As before, suppose that no two points (intersection points or endpoints) have the same x-coordinate.

The other difference is that here we want to construct a single cell only, the cell that contains x. Whenever a segment closes a face, splitting it into two, we discard one of the two resulting faces and keep only the one that contains x. To detect whether a face is closed, use a disjoint-set (union-find) data structure on S. Initially, all segments are in separate components.

Insert the segments of S in order $s_1, \ldots, s_n$, chosen uniformly at random. Maintain (as a doubly connected edge list) the trapezoidal decomposition of the face f_i of the arrangement $\mathcal{A}_i$ of $\{s_1, \ldots, s_i\}$ that contains x.

As a third data structure, maintain a *history dag* (directed acyclic graph) on all trapezoids that appeared at some point of the construction. For each trapezoid there, store the (at most four) segments that define it. The root of this dag corresponds to the entire plane and has no segments associated to it.

Those trapezoids that are part of the current face f_i appear as *active* leaves in the history dag. There are two more categories of vertices: Either the trapezoid was destroyed at some step by a segment crossing it; in this case, it is an interior vertex of the history dag and stores links to the (at most four) new trapezoids that replaced it. Or the trapezoid was cut off at some step by a segment that did not cross it but excluded it from the face containing x; these vertices are called *inactive* leaves and they will remain so for the rest of the construction.

Insertion of a segment s_{r+1} comprises the following steps.

1. Find the cells of the trapezoidal map f_r^* of f_r that s intersects by propagating s_{r+1} down the history dag.

2. Split and merge the cells found in Step 1. For each split, store the new trapezoids with the old one that is replaced.

   Wherever in a split s_{r+1} connects two segments s_j and s_k, join the components of s_j and s_k in the union find data structure. If they were in the same component already, then f_r is split into two faces. Determine which trapezoids are cut off from f_{r+1} at this point by alternately exploring both components using the DCEL structure. (Start two depth-first searches one each from the two local trapezoids incident to s_{r+1}. Proceed in both searches alternately until one is finished. Mark all trapezoids as discarded that are in the component that does not contain x.) In this way, the time spent for the exploration is proportional to the number of trapezoids discarded and every trapezoid can be discarded at most once.

3. Update the history dag using the information stored during splits. This is done only after all splits have been processed in order to avoid updating trapezoids that are discarded in this step.

The analysis is completely analogous to the case of disjoint segments, except for the expected number of trapezoids in f_r^*. By Theorem 14.2 there are $O(\lambda_3(n))$ edges bounding f_r and therefore also $O(\lambda_3(n))$ faces. Using the notation of the configuration space framework we obtain $t_{r+1} = E(|f_r^*|) = O(\lambda_3(r))$. The expected number of conflicts is bounded from above by

$$\begin{aligned}
\sum_{r=1}^{n-1}(k_1 - k_2 + k_3) &\leq 16(n-1) + 12n\sum_{r=1}^{n-1}\frac{\lambda_3(r+1)}{r(r+1)} \\
&\leq 16(n-1) + 12\sum_{r=1}^{n-1}\frac{n}{r+1}\lambda_3(r+1)\frac{1}{r} \\
&\leq 16(n-1) + 12\sum_{r=1}^{n-1}\frac{\lambda_3(n)}{r} \\
&= 16(n-1) + 12\,\lambda_3(n)H_{n-1} \\
&= O(\lambda_3(n)\log n)
\end{aligned}$$