Title:
Straight Skeletons and Mitered Offsets of Polyhedral Terrains in 3D

Authors:
Martin Held, held@cs.sbg.ac.at, FB Computerwissenschaften, University of Salzburg, Austria
Peter Palfrader, palfrader@cs.sbg.ac.at, FB Computerwissenschaften, University of Salzburg, Austria

Keywords:
Straight Skeleton, Monotone Surface, Mitered Offset, 3D

DOI: 10.14733/cadconfP.2018.xxx-yyy

Offsetting and Straight Skeletons:

Offsetting is a fundamental operation both in CAD as well as in several application areas. In a nutshell, offsetting a plane polygon $\mathcal{P}$ by a constant-radius offset with offset distance t requires us to determine all points of the plane that are within distance t from $\mathcal{P}$. The resulting offset curve will contain straight-line segments and circular arcs. A mitered offset is obtained by dropping the offset arcs and extending the offset segments in order to make them meet. For mitered offsets, an offset segment is not at a fixed distance to its source segment but instead to the line supporting the source segment.

Nowadays it is generally uncontested that computing an appropriate skeletal data structure as preprocessing constitutes the premier choice for offsetting polygons with regard to both speed and reliability. See, e.g., constant-radius offsets based on Voronoi diagrams [4] and mitered offsets based on straight skeletons [6]. Hence, it seems natural to apply a similar approach to mitered offsetting in 3D and to resort to 3D pendants of straight skeletons.

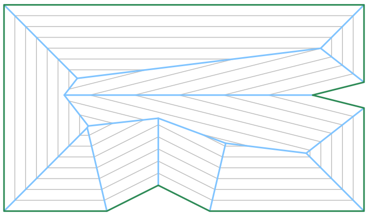

Fig. 1: The straight skeleton (blue) of an input polygon (green); several wavefronts are shown in gray.

Let $\mathcal{P}$ be a simple polygon in the plane and consider the following process [1]: At time $t := 0$, each edge of $\mathcal{P}$ starts to move towards the interior of $\mathcal{P}$ at unit speed in a self-parallel manner, thereby maintaining incidences. The set of moving edges forms one or more polygons, called the wavefront $\mathcal{W}_{\mathcal{P}}(t)$ of $\mathcal{P}$ at time t, see Fig. 1. Note that each edge of $\mathcal{W}_{\mathcal{P}}(t)$ is at all times at orthogonal distance t to its corresponding edge of $\mathcal{P}$.

The wavefront needs to be updated at times to remain a set of simple polygons: As edges shrink to zero length (*edge event*), they are removed, and edges are split and incidences updated when a previously non-incident vertex moves into their interior (*split event*). The straight skeleton $\mathcal{S}(\mathcal{P})$ of $\mathcal{P}$ is then defined as the geometric graph whose edges consist of the traces of wavefront vertices over the propagation process.

A mitered offset of $\mathcal{P}$ at offset distance t corresponds to the wavefront at time t. Mitered offsets are inherently linked to the straight skeleton: Given the straight skeleton $\mathcal{S}(\mathcal{P})$ of a polygon $\mathcal{P}$ with n vertices, any mitered offset can be constructed in $\mathcal{O}(n)$ time and space [6]. We note that obtaining a mitered offset of $\mathcal{P}$ based on $\mathcal{S}(\mathcal{P})$ is not just easy to implement but it is also reliable and numerically stable.

Moving to Three Dimensions

Straight skeletons of polyhedral objects in 3D were studied by Barequet et al. [3] and, more recently, by Aurenhammer and Walzl [2]. However, while combinatorial complexities have been established for the straight skeleton of polytopes, no runtime bounds have been investigated. Furthermore, it is not obvious how to implement Aurenhammer and Walzl's approach in full generality such that it can cope with arbitrary polyhedral objects in 3D.

In this work we consider straight skeletons and mitered offsets of polyhedral terrains in 3D. As usual, a polyhedral terrain $\mathcal{T}$ is a piecewise-linear, continuous function of two variables. E.g., triangulated irregular networks (TINs) known to GIS form polyhedral terrains. To simplify matters, we assume that $\mathcal{T}$ is defined over all of $\mathbb{R}^2$ and that all facets are simply-connected. Furthermore, we assume that $\mathcal{T}$ is in general position: No more than four supporting planes of the facets of $\mathcal{T}$ shall be tangent to a common sphere, and the degree of any vertex of $\mathcal{T}$ shall be at most a constant k.

Wavefront Propagation and Events:

We consider the wavefront propagation of a polyhedral terrain $\mathcal{T}$. Similar to the 2D setting, the wavefront consists of wavefront facets which are at orthogonal distance t to their corresponding input facets at all times. Initially, at time $t := 0$, the wavefront $\mathcal{W}_\mathcal{T}(t)$ is identical to $\mathcal{T}$. When the propagation process starts, all facets of the wavefront move upwards, in positive z-direction. During this propagation, incidences are retained where possible.

For the initial offset at time $t := \delta$, for any sufficiently small $\delta > 0$, retaining the combinatorial structure is possible along edges. Furthermore, locally at vertices of degree three, a mitered offset of the same combinatorial structure is possible. However, at vertices of degree four or more, any offset, even at an infinitesimally small δ, will generally have a combinatorial structure different from the input: The offset surface consists locally of several degree-three vertices that arise from the offsets of the planes incident at the input vertex of higher degree; see Aurenhammer and Walzl [2].

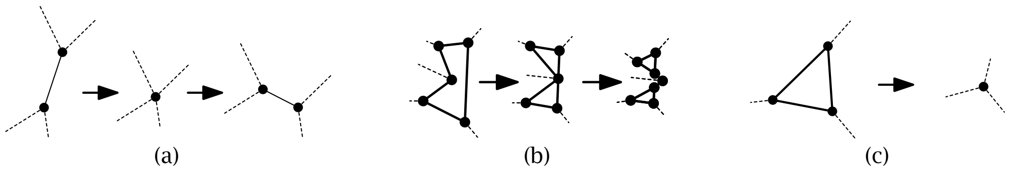

Fig. 2: Edge event (a), split event (b), and face event (c) during the wavefront propagation.

As the wavefront propagation continues, the combinatorial structure of the surface has to be updated and the set of wavefront vertices and their trajectories change at discrete points in time at so-called *events*, when four or more wavefront facets pass through a common point.

Aurenhammer and Walzl [2] differentiate between events that change the topology of the offset and events that merely change its geometry. However, since our wavefront surface is z-monotone and continuous, we will only observe the second class of events, *surface events*, in the wavefront propagation.

An *edge event* happens at time t when an edge of the wavefront collapses to zero length without its incident facets vanishing, too. The two vertices incident at the edge are merged, giving rise to a high-degree vertex. For the wavefront after the event, at time $t + \delta$, this high-degree vertex has to be resolved and generally split again similar to the process at the initial wavefront construction. See Fig. 2(a).

A *(facet) split event* happens when a vertex v of the wavefront that is incident at facet f moves into the interior of another edge e of f without f collapsing. This case is similar to the split event known from 2D straight skeletons. Combinatorially, the edge e is split at the locus of v and made incident to v, creating a higher-degree vertex which then needs to be resolved again for the post-event wavefront. See Fig. 2(b).

In the third event type, the *face event*, a facet f may collapse to an empty area. This coincides with one or more edges of f collapsing or a vertex of f moving into the interior of another edge of f. At the event time t, the facet is replaced by a set of edges that cover its boundary without overlapping, thereby merging vertices which now occupy the same locus (if such vertices exist). Again, for the post-event wavefront higher-degree vertices may need to be resolved. See Fig. 2(c).

The propagation of the wavefront is finished once no more events occur. The three dimensional straight skeleton $\mathcal{S}(\mathcal{T})$ of $\mathcal{T}$ is then the structure whose edges are the traces of wavefront vertices and whose facets are the traces of wavefront edges. To unambiguously refer to features of the 3D straight skeleton, Aurenhammer et al. [2] call the edges of $\mathcal{S}(\mathcal{T})$ *spokes* and its facets *sheets*. The volumes bounded by sheets are called *cells*.

Interior vertices correspond to events that have been observed in the propagation process. Any wavefront vertex or edge remaining in the wavefront at the end of the process induces an unbounded straight skeleton spoke or sheet which continues to infinity.

Simulating the Wavefront Propagation for Computing the Straight Skeleton:

We compute the straight skeleton $\mathcal{S}(\mathcal{T})$ of $\mathcal{T}$ by simulating its wavefront propagation. This requires determining at every stage in the process what the next event will be. To cope with this problem we maintain a priority queue of potential events: As initialization, we first create the initial wavefront for time $t = \delta$, where δ is infinitesimally small, splitting higher degree vertices of $\mathcal{T}$. Then we store for every edge of the wavefront its collapse time, and we store for every vertex of the wavefront the instances of when it will move into any of the edges of its incident facets.

To advance time in our simulation of the wavefront propagation, we fetch the event from the priority queue with the earliest time. We process this event by modifying the wavefront combinatorics according to the event type, thereby merging and then splitting vertices as described in the previous section. We add new events to the priority queue for all edges and vertices that were affected or created by the event.

Then we proceed and fetch the next item from the priority queue. We need to verify that it still is a valid event, that is, we need to check that the edge that is supposed to collapse or the vertex that is supposed to move into an edge are still elements of the wavefront — prior events may have already restructured the wavefront and invalidated this event. If it is a valid event then we process it as described. Otherwise we simply drop it. In either case, this process is repeated until the priority queue is empty.

Number of events

In general, an event happens at point p and time t when four (or more) wavefront facets become incident. (For simplicity reasons, our general position assumption guarantees that no more than four wavefront facets are involved in an event.) This provides a trivial upper bound of $\binom{n}{4}$ on the size of the priority queue, where n is the number of facets of the input surface. Based on our experience with different algorithms for computing straight skeletons in the plane, we conjecture that only a very small subset of those $\binom{n}{4}$ combinations will be relevant in practice.

Splitting higher-degree vertices

Aurenhammer and Walzl [2] show that an offset surface of a higher-degree vertex v always exists even though it is not necessarily unique. One offset that always exists corresponds to a wavefront where v has been replaced by a tree. In [2], they suggest as a simple approach to enumerate all combinatorially different trees and check whether they correspond to valid offset surfaces of v. The geometry of a tree's element is dictated by its combinatoric properties. Such a valid tree will replace the vertex v in the propagating wavefront. By our general position assumption, all vertices of the input surface have at most constant degree k. Thus, finding this tree for a single vertex v is a constant-time operation as well. Furthermore, at most a constant number of elements need to be added to the wavefront per input vertex.

Vertex degrees during events

After having constructed the initial wavefront, all moving vertices will be of degree three in the generic case. We investigate the types of vertices that can appear in events. In an edge event, the edge that connects to degree-three vertices collapses, giving rise to a degree-four vertex v, as shown in Fig. 2(a). In the generic case, v will have to be split (at constant cost) into two new degree-three vertices connected by a new edge. In our general position assumption we stated that no more than four supporting planes of faces may be tangent to a common sphere. Thus, for our input we will always either split v into two vertices, or v will never again participate in an event.

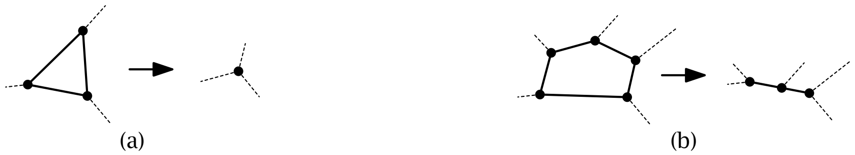

Fig. 3: Two types of face collapsing events.

In a split event, a degree-three vertex v comes to lie on previously non-incident wavefront edge e, which is split in two during the event, giving rise to a degree-five vertex (Fig. 2(b)). In the generic case this vertex will be split into three new vertices, each of degree three.

For face events we can distinguish two sub-types (Fig. 3). In one, a triangle facet will collapse as all its incident edges shrink to zero length. This will give immediate rise to a new degree-three vertex which can then propagate. The other type is where a more complex polygon collapses as some of its edges collapse and maybe some vertices become incident at other edges of the polygon. The facet is replaced by one or more edges, and all resulting vertices will be of degree three and can propagate without any need to be split.

Obtaining Offset Surfaces:

If only a single mitered offset surface at orthogonal distance t is sought, then one approach to construct this offset is to simply run the wavefront propagation process until time t. Then, the wavefront at this time is the offset surface sought. See Fig. 4 for a sample input and offset surface.

If multiple offset surfaces at different distances should be constructed or if the straight skeleton is already available, then we apply the following process to obtain an offset surface in time linear in the size of the straight skeleton: For a given spoke s, we denote by $s(t)$ the three dimensional point obtained by intersecting s with a plane at distance t and parallel to the base of any one of its incident cells.

For every spoke s of the skeleton which exists at (orthogonal) offsetting distance t, i.e., for which $s(t)$ exists, and for every cell c incident at s where (s, c) has not been processed before, we construct an offset facet as follows: Let f_1 be one of the sheets of $s_1 := s$ that is on the boundary of c. We walk along the boundary of f_1, moving in the direction of positive z, until we reach another spoke s_2 of f_1 that exists at distance t. Now let f_2 be the other sheet of c incident at s_2 and repeat the walk in f_2 to find a spoke s_3. Eventually, we will return to our initial spoke s_1. Let s_ℓ be the last one before we returned. (Special handling will be required to process the case of infinite elements.) The polygon with vertices $s_1(t), s_2(t), \ldots, s_\ell(t)$ is now a valid offset facet and we add it to the offset surface we are constructing. We then mark $(s_1, c), (s_2, c), \ldots, (s_\ell, c)$ as processed and continue with our main loop. Once all spokes have been processed, the set of offset facets represents the complete offset surface.

The correctness of this approach hinges on the property that all offset facets are simple polygons and contain no holes. This property stems from the fact that the wavefront propagation does not experience any piercing event since $\mathcal{T}$ is a terrain.

Discussion:

Wavefront propagation yields a simple algorithm to compute the straight skeleton of a terrain. The processing cost of each event is constant for generic input, and a (worst-case) bound for the number of events is $\binom{n}{4}$. Likely, this bound is far too loose but no tighter bound is known to theory. Maintaining the events in a priority queue results in a (pessimistic) runtime bound of $\mathcal{O}(n^4 \log n)$. A general $O(n^2)$ upper bound on the combinatorial complexity of $\mathcal{S}(\mathcal{T})$ is established by Aurenhammer and Walzl [2]. A construction by Held [5] yields a matching worst-case $\Omega(n^2)$ lower bound even for a terrain $\mathcal{T}$. His construction can be adapted to yield the same bound for the combinatorial complexity of a mitered offset.

For descriptive simplicity, our general-position assumption bounds the maximum degree of a vertex that may appear in the propagating wavefront by a small constant. However, using larger constants does not change the process significantly and only results in more complex event handling requirements.

Furthermore, we can relax the bound on the maximum degree of vertices of the input surface. Resolving higher-degree vertices where the degrees are not bound by a constant for the initial wavefront will require more than constant work, but at least for pointed vertices, where all incident faces are confined to one half space, offsetting can be reduced to computing weighted 2D straight skeletons [3] for which implementations exist [6]. Vertices that are saddle-points can still be handled by one of the methods described by Aurenhammer and Walzl [2].

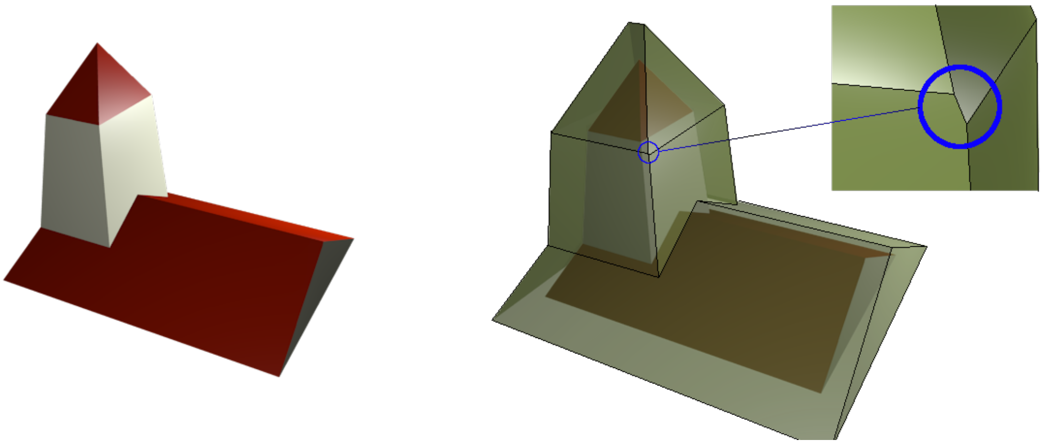

Fig. 4: (Left) Steeple with a simple roof as a polyhedral terrain. (Right) One sample mitered offset of this terrain overlayed transparently. Note that terrain vertices of degree larger than three split into multiple vertices in the offset (see detail).

Acknowledgments:
This work was supported by Austrian Science Fund (FWF): P25816-N15 and ORD 53-VO.

References:

[1] Aichholzer, O.; Aurenhammer, F.; Alberts, D.; Gärtner, B.: A Novel Type of Skeleton for Polygons. Journal of Universal Computer Science, 1(12), 1995, 752–761. doi:10.1007/978-3-642-80350-5_65.

[2] Aurenhammer, F.; Walzl, G.: Straight Skeletons and Mitered Offsets of Nonconvex Polytopes. Discrete & Computational Geometry, 56(3), 2016, 743–801. doi:10.1007/s00454-016-9811-5.

[3] Barequet, G.; Eppstein, D.; Goodrich, M. T.; Vaxman, A.: Straight Skeletons of Three-Dimensional Polyhedra. 16th Annual European Symposium on Algorithms, Lecture Notes in Computer Science, volume 5193. Springer-Verlag, 2008, 148–160. doi:10.1007/978-3-540-87744-8_13.

[4] Held, M.: On the Computational Geometry of Pocket Machining, Lecture Notes in Computer Science, volume 500. Springer-Verlag, 1991. ISBN 3-540-54103-9.

[5] Held, M.: On Computing Voronoi Diagrams of Convex Polyhedra by Means of Wavefront Propagation. 6th Canadian Conference on Computational Geometry. 1994, 128–133.

[6] Palfrader, P.; Held, M.: Computing Mitered Offset Curves Based on Straight Skeletons. Computer-Aided Design and Applications, 12(4), 2015, 414–424. doi:10.1080/16864360.2014.997637.