58th Annual IEEE Symposium on Foundations of Computer Science

Removing Depth-Order Cycles Among Triangles: An Efficient Algorithm Generating Triangular Fragments

Mark de Berg
Department of Computer Science
TU Eeindhoven
Eindhoven, the Netherlands
Email: m.t.d.berg@tue.nl

***Abstract*—More than 25 years ago, inspired by applications in computer graphics, Chazelle *et al.* (FOCS 1991) studied the following question: Is it possible to cut any set of n lines or other objects in $\mathbb{R}^3$ into a subquadratic number of fragments such that the resulting fragments admit a depth order? They managed to prove an $O(n^{9/4})$ bound on the number of fragments, but only for the very special case of bipartite weavings of lines. Since then only little progress was made, until a recent breakthrough by Aronov and Sharir (STOC 2016) who showed that $O(n^{3/2} \operatorname{polylog} n)$ fragments suffice for any set of lines. In a follow-up paper Aronov, Miller and Sharir (SODA 2017) proved an $O(n^{3/2+\varepsilon})$ bound for triangles, but their method uses high-degree algebraic arcs to perform the cuts. Hence, the resulting pieces have curved boundaries. Moreover, their method uses polynomial partitions, for which currently no algorithm is known. Thus the most natural version of the problem is still wide open: Is it possible to cut any collection of n disjoint triangles in $\mathbb{R}^3$ into a subquadratic number of triangular fragments that admit a depth order? And if so, can we compute the cuts efficiently?**

We answer this question by presenting an algorithm that cuts any set of n disjoint triangles in $\mathbb{R}^3$ into $O(n^{7/4} \operatorname{polylog} n)$ triangular fragments that admit a depth order. The running time of our algorithm is $O(n^{3.69})$. We also prove a refined bound that depends on the number, K, of intersections between the projections of the triangle edges onto the xy-plane: we show that $O(n^{1+\varepsilon} + n^{1/4}K^{3/4} \operatorname{polylog} n)$ fragments suffice to obtain a depth order. This result extends to xy-monotone surface patches bounded by a constant number of bounded-degree algebraic arcs in general position, constituting the first subquadratic bound for surface patches. Finally, as a byproduct of our approach we obtain a faster algorithm to cut a set of lines into $O(n^{3/2} \operatorname{polylog} n)$ fragments that admit a depth order. Our algorithm for lines runs in $O(n^{5.38})$ time, while the previous algorithm uses $O(n^{8.77})$ time.

***Keywords*-computational geometry; depth orders; cyclic overlap;**

I. Introduction

A. Motivation and problem statement

Let T and T' be two disjoint triangles (or other objects) in $\mathbb{R}^3$. We that T is *below* T'—or, equivalently, that T' is *above* T—when there is a vertical line ℓ intersecting both T and T' such that $\ell \cap T$ has smaller z-coordinate than $\ell \cap T'$. We denote this relation by $T \prec T'$. Note that two triangles may be unrelated by the $\prec$-relation, namely when their vertical projections onto the xy-plane are disjoint. Now let $\mathcal{T}$ be a collection of n disjoint triangles in $\mathbb{R}^3$. A *depth order* (for the vertical direction) on $\mathcal{T}$ is a total order on $\mathcal{T}$ that is consistent with the $\prec$-relation, that is, an ordering $T_1, \ldots, T_n$ of the triangles such that $T_i \prec T_j$ implies $i < j$.

Depth orders play an important role in many applications. For example, the Painter's Algorithm from computer graphics performs hidden-surface removal by rendering the triangles forming the objects in a scene one by one, drawing each triangle "on top of" the already drawn ones. To give the correct result the Painter's Algorithm must handle the triangles in depth order with respect to the viewing direction. Several object-space hidden-surface removal algorithms and ray-shooting data structures need a depth order as well. Depth orders also play a role when one wants to assemble a product by putting its constituent parts one by one into place using vertical translations [23]. The problem of computing a depth order for a given set of objects has therefore received considerable attention [2], [7], [10], [11]. However, a depth order does not always exist since there can be *cyclic overlap*, as illustrated in Fig. 1(i). In such cases the algorithms above simply report that no depth exists. What we would then like to do is to cut the triangles into fragments such that the resulting set of fragments is acyclic (that is, admits a depth order). This gives rise to the following problem: How many fragments are needed in the worst case to ensure that a depth order exists? And how efficiently can we compute a set of cuts resulting in a small set of fragments admitting a depth order?

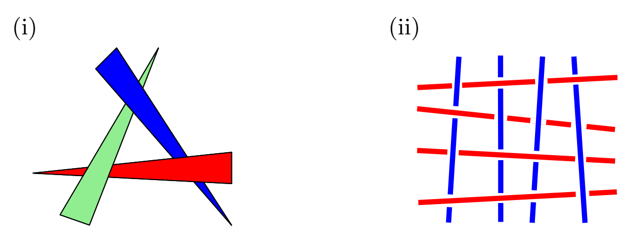

Figure 1. (i) Three triangles with cycle overlap. (ii) A bipartite weaving.

0272-5428/17 $31.00 © 2017 IEEE
DOI 10.1109/FOCS.2017.33

B. Previous work

The problem of bounding the worst-case number of fragments needed to remove all cycles from the depth-order relation has a long history. In the special case of lines (or line segments) one can easily get rid of all cycles using $O(n^2)$ cuts: project the lines onto the xy-plane and cut each line at its intersection points with the other lines. A lower bound on the worst-case number of cuts is $\Omega(n^{3/2})$ [15]. It turned out to be amazingly hard to get any subquadratic upper bound. In 1991 Chazelle *et al.* [15] obtained such a bound, but only for so-called bipartite weavings; see Fig. 1(ii). Moreover, their $O(n^{9/4})$ bound is still far away from the $\Omega(n^{3/2})$ lower bound. Later Aronov *et al.* [5] obtained a subquadratic upper bound for general sets of lines, but they only get rid of all triangular cycles—that is, cycles consisting of three lines—and their bounds are only slightly subquadratic: they use $O(n^{2-1/69}\log^{16/69} n)$ cuts to remove all triangular cycles. (They obtained a slightly better bound of $O(n^{2-1/34}\log^{8/17} n)$ for removing all so-called elementary triangular cycles.) Finally, several authors studied the algorithmic problem of computing a minimum-size complete cut set—a *complete cut set* is a set of cuts that removes all cycles from the depth-order relation—for a set of lines (or line segments). Solan [21] and Har-Peled and Sharir [17] gave algorithms that produce a complete cut set of size roughly $O(n\sqrt{\text{OPT}})$, where OPT is the minimum size of any complete cut set for the given lines. Aronov *et al.* [3] showed that computing a minimum-size complete cut set is NP-hard, and they presented an algorithm that computes a complete cut set of size $O(\text{OPT}\cdot\log\text{OPT}\cdot\log\log\text{OPT})$ in $O(n^{4+2\omega}\log^2 n) = O(n^{8.764})$ time, where $\omega < 2.373$ is the exponent of the best matrix-multiplication algorithm.

Eliminating depth cycles from a set of triangles is even harder than it is for lines. The trivial bound on the number of fragments is $O(n^3)$; this bound can for instance be obtained by taking, for each triangle edge, a vertical cutting plane containing that edge. Paterson and Yao [19] showed already in 1990 that any set of disjoint triangles admits a so-called *binary space partition* (BSP) of size $O(n^2)$, which immediately implies an $O(n^2)$ bound on the number of fragments needed to remove all cycles. Indeed, a BSP ensures that the resulting set of triangle fragments is acyclic for any direction, not just for the vertical direction. Better bounds on the size of BSPs are known for fat objects (or, more generally, low-density sets) [8] and for axis-aligned objects [1], [20], [22], but for arbitrary triangles there is an $\Omega(n^2)$ lower bound on the worst-case size of a BSP [13]. Hence, to get a subquadratic bound on the number of fragments needed to obtain a depth order, one needs a different approach.

In 2016, using Guth's polynomial partitioning technique [16], Aronov and Sharir [4] achieved a breakthrough in the area by proving that any set of n lines in $\mathbb{R}^3$ in general position can be cut into $O(n^{3/2}\operatorname{polylog} n)$ fragments such that the resulting set of fragments admits a depth order. A complete cut set of size $O(n^{3/2}\operatorname{polylog} n)$ can then be computed using the algorithm of Aronov *et al.* [3] mentioned above. They also gave a more refined bound for line segments, which depends on the number of intersections, K, between the segments in the projection. More precisely, they show that $O(n + n^{1/2}K^{1/2}\operatorname{polylog} n)$ cuts suffice. In a follow-up paper, Aronov, Miller and Sharir [6] extended the result to triangles: they show that, for any fixed $\varepsilon > 0$, any set of disjoint triangles in general position can be cut into $O(n^{3/2+\varepsilon})$ fragments that admit a depth order. This may seem to almost settle the problem for triangles, but the result of Aronov, Miller and Sharir has two serious drawbacks.

- The technique does not result in triangular fragments, since it cuts the triangles using algebraic arcs. The degree of these arcs is exponential in the parameter ε appearing in $O(n^{3/2+\varepsilon})$ bound.
- The technique, while being in principle constructive, does not give an efficient algorithm, since currently no algorithms are known for constructing Guth's polynomial partitions.

Arguably, the natural way to pose the problem for triangles is that one requires the fragments to be triangular as well—polygonal fragments can always be decomposed further into triangles, without increasing the number of fragments asymptotically—so especially the first drawback is a major one. Indeed, Aronov, Miller and Sharir state that "*It is a natural open problem to determine whether a similar bound can be achieved with straight cuts [...]. Even a weaker bound, as long as it is subquadratic and generally applicable, would be of great significance.*" Another open problem stated by Aronov, Miller and Sharir is to extend the result to surface patches: "*Extending the technique to curved objects (e.g., spheres or spherical patches) is also a major challenge.*"

C. Our contribution

We prove that any set $\mathcal{T}$ of n disjoint triangles in $\mathbb{R}^3$ can be cut into $O(n^{7/4}\operatorname{polylog} n)$ triangular fragments that admit a depth order. Thus we overcome the first drawback of the method of Aronov, Miller and Sharir, although admittedly our bound is not as sharp as theirs. We also overcome the second drawback, by presenting an algorithm to perform the cuts in $O(n^{5/2+\omega/2}\log^2 n) = O(n^{3.69})$ time. Here $\omega < 2.373$ is, as above, the exponent of the best matrix-multiplication algorithm. As a byproduct, we improve the time to compute a complete cut set of size $O(n^{3/2}\operatorname{polylog} n)$ for a collection of lines: we show that a simple trick reduces the running time from $O(n^{4+2\omega}\log^2 n)$ to $O(n^{3+\omega}\log^2 n)$.

We also present a more refined approach that yields a bound of $O(n^{1+\varepsilon} + n^{1/4}K^{3/4}\operatorname{polylog} n)$ on the number of fragments, where K is the number of intersections between the triangles in the projection. This result extends to xy-monotone surface patches bounded by a constant number of bounded-degree algebraic arcs in general position. Thus we make progress on all open problems posed by Aronov, Miller and Sharir.

Finally, as a minor contribution we get rid of the non-degeneracy assumptions that Aronov and Sharir [4] make when eliminating cycles from a set of segments. Most degeneracies can be handled by a straightforward perturbation argument, but one case—parallel segments that overlap in the projection—requires some new ideas. Being able to handle degeneracies for segments implies that our method for triangles can handle degeneracies as well.

II. Eliminating cycles among triangles

A. Overview of the method

We first prove a proposition that gives conditions under which the existence of a depth order for a set of triangles is implied by the existence of a depth order for the triangle edges. The idea is then to take a complete cut set for the triangle edges—there is such a cut set of size $O(n^{3/2} \operatorname{polylog} n)$ by the results of Aronov and Sharir—and "extend" the cuts (by taking vertical planes through the cut points) so that the conditions of the proposition are met. A straightforward extension would generate too many triangle fragments, however. Therefore our cutting procedure has two phases. In the first phase we localize the problem by partitioning space into regions such that (i) the collection of regions admits a depth order, and (ii) each region is intersected by only few triangles. (This localization is also the key to speeding up the algorithm for lines.) In the second phase we then locally (that is, inside each region) extend the cuts from a complete cut set for the edges, so that the conditions of the proposition are met.

B. Notation and terminology

Let $\mathcal{T}$ denote the given set of disjoint non-vertical triangles, let $\mathcal{E}$ denote the set of edges of the triangles in $\mathcal{T}$, and let $\mathcal{V}$ denote the set of vertices of the triangles. We assume the triangles in $\mathcal{T}$ are closed. However, at the places where a triangle is cut it becomes open. Thus the edges of a triangle fragment that are (parts of) edges in $\mathcal{E}$ are part of the fragment, while edges that are induced by cuts are not. We denote the vertical projection of an object o in $\mathbb{R}^3$ onto the xy-plane by $\overline{o}$.

C. Relating depth orders for edges to depth orders for triangles

We define a *column* to be a 3-dimensional region $C_\Delta := \Delta \times (-\infty, +\infty)$, where Δ is an open convex polygon on the xy-plane. Our cutting procedure is based on the following proposition.

Proposition 1: Let $\mathcal{T}$ be a set of disjoint triangles in $\mathbb{R}^3$, and let $\mathcal{E}$ be the set of edges of the triangles in $\mathcal{T}$. Let C_Δ be a column whose interior does not contain a vertex of any triangle in $\mathcal{T}$, and let $\mathcal{T}_\Delta := \{T \cap C_\Delta : T \in \mathcal{T}\}$ and $\mathcal{E}_\Delta := \{e \cap C : e \in \mathcal{E}\}$. Then $\mathcal{T}_\Delta$ admits a depth order if $\mathcal{E}_\Delta$ admits a depth order.

Proof: For a triangle $T_i \in \mathcal{T}$, define $P_i := T_i \cap C$. Thus $\mathcal{T}_\Delta = \{P_i : T_i \in \mathcal{T}\}$. Assume $\mathcal{E}_\Delta$ admits a depth order and suppose for a contradiction that $\mathcal{T}_\Delta$ does not.

Consider a cycle $\mathcal{C} := P_0 \prec P_1 \prec \cdots \prec P_{k-1} \prec P_0$ in $\mathcal{T}_\Delta$. As observed by Aronov *et al.* [6] we can associate a closed curve in $\mathbb{R}^3$ to $\mathcal{C}$, as follows. For each pair P_i, P_{i+1} of consecutive polygons in $\mathcal{C}$—here and in the rest of the proof indices are taken modulo k—let $b_i \in P_i$ and $a_{i+1} \in P_{i+1}$ be points such that the segment $b_i a_{i+1}$ is vertical. We refer to the closed polygonal curve whose ordered set of vertices is $a_0, b_0, a_1, b_1, \ldots, a_{k-1}, b_{k-1}$ as a *witness curve* for $\mathcal{C}$. We call the vertical segments $b_i a_{i+1}$ the *connections* of $\Gamma(\mathcal{C})$, and we call the segments $a_i b_i$ the *links* of $\Gamma(\mathcal{C})$. Since the connections are vertical, we have $\overline{a_{i+1}} = \overline{b_i}$ and so we can write $\overline{\Gamma(\mathcal{C})}$ as $\overline{a_0}, \overline{a_1}, \ldots, \overline{a_{k-1}}, \overline{a_0}$. Note that $\overline{a_i} \in \overline{P_{i-1}} \cap \overline{P_i}$ for all i. In general, the points a_i and b_i can be chosen in many ways and so there are many possible witness curves. We will need a specific witness curve, as specified next. We say that a link $a_i b_i$ is *good* if a_i and b_i lie on the same edge of their polygon P_i—this edge is also an edge in $\mathcal{E}_\Delta$—and we say that $a_i b_i$ is *bad* otherwise. We now define the *weight* of a witness curve Γ to be the number of bad links in Γ, and we define $\Gamma(\mathcal{C})$ to be any minimum-weight witness curve for $\mathcal{C}$.

Now consider a minimal cycle $\mathcal{C}^* := P_0 \prec P_1 \prec \cdots \prec P_{k-1} \prec P_0$ in $\mathcal{T}_\Delta$. (A cycle is minimal if any strict subset of polygons from the cycle is acyclic.) We will argue that we can find a cycle in $\mathcal{E}_\Delta$ consisting of edges of the polygons in $\mathcal{C}^*$, thus contradicting that $\mathcal{E}_\Delta$ admits a depth order.

Claim. All links $a_i b_i$ of $\Gamma(\mathcal{C}^*)$ are good.

Proof. Consider any link $a_i b_i$. Observe that $\overline{a_{i-1}}$ and $\overline{a_{i+2}}$ must both lie outside $\overline{P_i}$, otherwise $\mathcal{C}^*$ is not minimal. Consider $\Delta \setminus \overline{P_i}$, the complement of $\overline{P_i}$ inside the column base Δ. The region $\Delta \setminus \overline{P_i}$ consists of one or more connected components. (It cannot be empty since then P_i cannot be part of any cycle in $\mathcal{T}_\Delta$.) Each connected component is separated from $\overline{P_i}$ by a single edge of $\overline{P_i}$, since by assumption T_i does not have a vertex inside C and so $\overline{P_i}$ does not have a vertex inside Δ either. We now consider two cases, as illustrated in Fig. 2.

Case I: $\overline{a_{i-1}}$ and $\overline{a_{i+2}}$ lie in different components of $\Delta \setminus \overline{P_i}$. Let $\overline{p}$ be the point where $\overline{a_{i-1} a_i}$ enters $\overline{P_i}$. Let $p \in P_i$ project onto $\overline{p}$ and let e be the edge of P_i containing p. (Possibly $\overline{p} = \overline{a_i}$.) Since $\overline{a_{i+2}}$ lies in a different connected component of $\Delta \setminus \overline{P_i}$ than $\overline{a_{i-1}}$, the projection $\overline{\Gamma(\mathcal{C}^*)}$ must cross $\overline{e}$ a second time, at some point $\overline{q}$. This leads to a contradiction with the minimality of $\mathcal{C}^*$. To see this, let $q \in \Gamma(\mathcal{C}^*)$ be a point

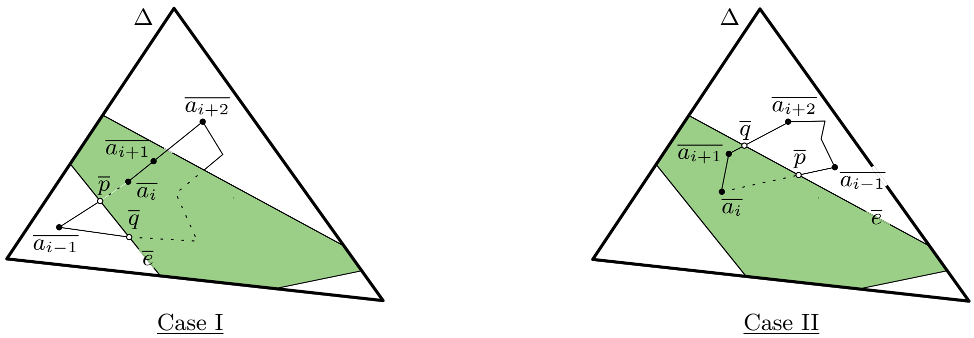

Figure 2. The two cases in the proof of Proposition 1. Polygon $\overline{P_i}$ is shown in green.

projecting onto $\overline{q}$ and let P_j be such that $q \in P_j$. Then $j \notin \{i-1, i, i+1\}$, because $a_{i-1}b_{i-1}$, and $a_i b_i$, and $a_{i+1}b_{i+1}$ are the only links of $\Gamma(\mathcal{C}^*)$ on P_{i-1}, and P_i, and P_{i+1}, respectively. But since $\overline{P_i} \cap \overline{P_j} \neq \emptyset$ we have $P_i \prec P_j$ or $P_j \prec P_i$, and so $j \notin \{i-1, i, i+1\}$ contradicts that $\mathcal{C}^*$ is minimal. Thus Case I cannot happen.

Case II: $\overline{a_{i-1}}$ and $\overline{a_{i+2}}$ lie in the same component of $\Delta \setminus \overline{P_i}$. In this case $a_i b_i$ must be a good link, because a_i and b_i must both lie on the edge e bordering the component of $\Delta \setminus \overline{P_i}$ that contains $\overline{a_{i-1}}$ and $\overline{a_{i+2}}$. Indeed, if $\overline{a_i}$ and/or $\overline{a_{i+1}}$ would not lie on e then we can obtain a witness curve of lower weight for $\mathcal{C}^*$, namely if we replace a_i by the point p such that $\overline{p} = \overline{a_{i-1}a_i} \cap \overline{e}$ and we replace a_{i+1} by the point q such that $\overline{q} = \overline{a_1 a_{i+1}} \cap \overline{e}$. (Note that if $a_{i-1} = a_{i+2}$, which happens when $\mathcal{C}^*$ consists of only three polygons, then the argument still goes through.)

Thus $a_i b_i$ is a good link, as claimed. ◇

If all links a_i, b_i of $\Gamma(\mathcal{C}^*)$ are good then $\mathcal{C}^*$ gives a cycle in $\mathcal{E}_\Delta$, contradicting that $\mathcal{E}(\sigma)$ admits a depth order. Hence, the assumption that $\mathcal{T}_\Delta$ contains a cycle is false. ■

D. The cutting procedure

A naive way to apply Proposition 1 would be the following: compute a complete cut set X for the set $\mathcal{E}$ of triangle edges, and take a vertical plane parallel to the yz-plane through each point in $\mathcal{V} \cup X$. This subdivides $\mathbb{R}^3$ into columns C_Δ (where each column base Δ is an infinite strip). These columns do not contain triangle vertices and the edge fragments inside each column are acyclic, and so the triangle fragments we obtain are acyclic. Unfortunately this straightforward approach generates too many fragments. Hence, we first subdivide space such that we do not cause too much fragmentation when we take the vertical planes through $\mathcal{V} \cup X$. The crucial idea is to create the subdivision based on the projections of the triangle edges. This allows us to use an efficient 2-dimensional partitioning scheme resulting in cells that are intersected by only few projected triangles edges. The 2-dimensional subdivision will then be extended into $\mathbb{R}^3$, to obtain 3-dimensional regions in which we can take vertical planes through $\mathcal{V} \cup X$ without creating too many fragments. We cannot completely ignore the triangles themselves, however, when we extend the 2-dimensional subdivision into $\mathbb{R}^3$—otherwise we already create too many fragments in this phase. Thus we create a hierarchical 2-dimensional subdivision, and we use the hierarchy to avoid cutting the input triangles into too many fragments. Next we make these ideas precise.

Let L be a set of n lines in the plane. A $(1/r)$-*cutting* for L is a partition Ξ of the plane into triangular[1] cells such that the interior of any cell $\Delta \in \Xi$ intersects at most n/r lines from L. We say that a cutting Ξ *c-refines* a cutting Ξ', where c is some constant, if every cell $\Delta \in \Xi$ is contained in a unique parent cell $\Delta' \in \Xi'$, and each cell in Ξ' contains at most c cells from Ξ. An *efficient hierarchical* $(1/r)$-*cutting* for L [18] is a sequence $\Psi := \Xi_0, \Xi_1, \ldots, \Xi_k$ of cuttings such that there are constants c, ρ such that the following four conditions are met:

(i) $\rho^{k-1} < r \leqslant \rho^k$,
(ii) Ξ_0 is the single cell $\mathbb{R}^2$,
(iii) Ξ_i is a $(1/\rho^i)$-cutting for L of size $O(\rho^{2i})$, for all $0 \leqslant i \leqslant k$,
(iv) Ξ_i is a c-refinement of Ξ_{i-1}, for all $1 \leqslant i \leqslant k$.

It is known that for any set L and any parameter r with $1 \leqslant r \leqslant n$, an efficient hierarchical $(1/r)$-cutting exists and can be computed in $O(nr)$ time [14], [18]. We can view Ψ as a tree in which each node u at level i corresponds to a cell $\Delta_u \in \Xi_i$, and a node v at level i is the child of a node u at level $i-1$ if $\Delta_v \subseteq \Delta_u$.

Our cutting procedure now proceeds in two steps. Recall that $\mathcal{T}$ denotes the given set of n triangles in $\mathbb{R}^3$, and $\mathcal{E}$ the set of $3n$ edges of the triangles in $\mathcal{T}$.

1) We start by constructing an efficient hierarchical $(1/r)$-cutting for L, with $r = n^{3/4}$, where L is the set of lines containing the edges in $\overline{\mathcal{E}}$. Next we cut the

[1] The cells may be unbounded, that is, we also allow wedges, half-planes, and the entire plane, as cells.

projection $\overline{T}$ of each triangle $T \in \mathcal{T}$ into pieces. This is done by executing the following recursive process on Ψ, starting at its root. Suppose we reach a node u of the tree. If $\Delta_u \subseteq \overline{T}$ or u is a leaf, then $\Delta_u \cap \overline{T}$ is one of the pieces of $\overline{T}$. Otherwise, we recursively visit all children v of u such that $\Delta_v \cap \overline{T} \neq \emptyset$.

After cutting each projected triangle $\overline{T}$ in this manner, we cut the original triangles $T \in \mathcal{T}$ accordingly. Let $\mathcal{T}_1$ denote the resulting collection of polygonal pieces. We extend the 2-dimensional cutting Ξ_k into $\mathbb{R}^3$ by erecting vertical walls through each of the edges in Ξ_k. Thus we create a column $C_\Delta := \Delta \times (-\infty, \infty)$ for each cell $\Delta \in \Xi_k$. Next, we cut each column C_Δ into vertical prisms by slicing it with each triangle $T \in \mathcal{T}$ that completely cuts through C_Δ, that is, we slice the column with each triangle T such that $\Delta \subseteq \overline{T}$. Let $\mathcal{S}$ denote the resulting 3-dimensional subdivision.

2) For each prism σ in the subdivision $\mathcal{S}$, we proceed as follows. Let $\mathcal{T}_1(\sigma) \subseteq \mathcal{T}_1$ be the set of pieces that have an edge intersecting the interior of σ, and let $\mathcal{E}(\sigma) := \{e \cap \sigma : e \in \mathcal{E}\}$. Note that $\mathcal{E}(\sigma)$ is the set of edges of the pieces in $\mathcal{T}_1(\sigma)$, where we only take the edges in the interior of σ. Let $X(\sigma)$ be a complete cut set for $\mathcal{E}(\sigma)$, and let $\mathcal{V}(\sigma) \subseteq \mathcal{V}$ be the set of triangle vertices in the interior of σ. For each point $q \in X(\sigma) \cup \mathcal{V}(\sigma)$, we take a plane $h(q)$ containing q and parallel to the yz-plane. Let $H(\sigma)$ be the resulting set of planes. We cut every piece $P \in \mathcal{T}_1(\sigma)$ into fragments using the planes in $H(\sigma)$.

We denote the set of fragments generated in Step 2 inside a prism σ by $\mathcal{T}_2(\sigma)$, and we denote the set of pieces in $\mathcal{T}_1$ that do not have an edge crossing the interior of any prism $\sigma \in \mathcal{S}$ by $\mathcal{T}_1^*$. (Note that $\mathcal{T}_1^*$ contains all pieces generated at internal nodes of Ψ.) Then $\mathcal{T}_2 := \mathcal{T}_1^* \cup \bigcup_{\sigma \in \mathcal{S}} \mathcal{T}_2(\sigma)$ is our final set of fragments.

Lemma 2: The set $\mathcal{T}_2$ of triangle fragments resulting from the procedure above is acyclic, and the size of $\mathcal{T}_2$ is $O(n^{7/4} + |X| \cdot n^{1/4})$, where $X := \bigcup_{\sigma \in \mathcal{S}} X(\sigma)$.

Proof: To prove that $\mathcal{T}_2$ is acyclic, define $\mathcal{S}^*$ to be the set of (open) prisms in $\mathcal{S}$, and consider the set $\mathcal{S}^* \cup \mathcal{T}_1^*$. We need the following claim.

Claim. The set $\mathcal{S}^* \cup \mathcal{T}_1^*$ admits a depth order.

Proof. By construction, for any object $o_i \in \mathcal{S}^* \cup \mathcal{T}_1^*$ there is a node $u \in \Psi$ such that $\overline{o_i} = \Delta_u$. Hence, for any two objects $o_1, o_2 \in \mathcal{S}^* \cup \mathcal{T}_1^*$ we have

$$\overline{o_1} \subseteq \overline{o_2}, \quad \text{or} \quad \overline{o_2} \subseteq \overline{o_1}, \quad \text{or} \quad \overline{o_1} \cap \overline{o_2} = \emptyset. \quad (1)$$

This implies that $\mathcal{S}^* \cup \mathcal{T}_1^*$ is acyclic. Indeed, suppose for a contradiction that $\mathcal{S}^* \cup \mathcal{T}_1^*$ does not admit a depth order. Consider a minimal cycle $\mathcal{C}^* := o_1 \prec o_2 \prec \cdots \prec o_k \prec o_1$. Obviously $k \geqslant 3$. But then (1) implies that we can remove o_1 or o_2 and still have a cycle, contradicting the minimality of $\mathcal{C}^*$. ◇

The claim above implies that $\mathcal{T}_2$ is acyclic if each of the sets $\mathcal{T}_2(\sigma)$ is acyclic. To see that $\mathcal{T}_2(\sigma)$ is acyclic, note that the planes in $H(\sigma)$ partition σ into subcells that do not contain a point from $X(\sigma)$ in their interior. Hence the set of edges of the fragments in such a subcell is acyclic—if this were not the case, then there would be a cycle left in $\mathcal{E}(\sigma)$, contradicting that $X(\sigma)$ is a complete cut set for $\mathcal{E}(\sigma)$. Moreover, a subcell does not contain any point from $\mathcal{V}(\sigma)$ in its interior, and so it does not contain a vertex of any fragment in its interior. We can therefore use Proposition 1 to conclude that within each subcell, the fragments are acyclic; the fact that the subcell is strictly speaking not a column—it may be bounded from above and/or below by a piece in $\mathcal{T}_1^*$—clearly does not invalidate the conclusion. Since the fragments in each subcell of σ are acyclic and the subcells are separated by vertical planes, $\mathcal{T}_2(\sigma)$ must be acyclic.

It remains to prove that $|\mathcal{T}_2| = O(n^{7/4} + |X| \cdot n^{1/4})$. We start by bounding $|\mathcal{T}_1|$. To this end, consider a triangle $T \in \mathcal{T}$ and let $P \in \mathcal{T}_1$ be a piece generated for T in Step 1. Let v be the node in Ψ where P was created. Then the cell Δ_u of the parent u of v is intersected by an edge of $\overline{T}$. Since each node in Ψ has $O(1)$ children and each cell $\Delta \in \Xi_i$ intersects at most n/ρ^i projected triangle edges, this means that

$$\begin{aligned} |\mathcal{T}_1| &= O\left(\sum_{i=0}^{k-1} \sum_{\Delta \in \Xi_i} n/\rho^i\right) \\ &= O\left(\sum_{i=0}^{k-1} \rho^{2i} \cdot (n/\rho^i)\right) \\ &= O(n\rho^k) \\ &= O(nr) \\ &= O(n^{7/4}). \end{aligned}$$

The number of additional fragments created in Step 2 can be bounded by observing that each prism σ in the subdivision $\mathcal{S}$ intersects at most $n/r = O(n^{1/4})$ triangle edges, and so $|T_1(\sigma)| = O(n^{1/4})$. If we now sum the number of additional fragments over all prisms σ in the subdivision $\mathcal{S}$ we obtain

$$\begin{aligned} &\text{number of additional fragments created in Step 2} \\ &\leqslant \sum_{\sigma \in \mathcal{S}} |H(\sigma)| \cdot |\mathcal{T}_1(\sigma)| \\ &\leqslant O(n^{1/4}) \cdot \left(\sum_{\sigma \in \mathcal{S}} |X(\sigma)| + \sum_{\sigma \in \mathcal{S}} |\mathcal{V}(\sigma)|\right) \\ &= O(n^{1/4}(|X| + n)). \end{aligned}$$

■

Lemma 2 leads to the following result.

Corollary 3: Suppose that any set of n lines has a complete cut set of size $\gamma(n)$. Then any set $\mathcal{T}$ of n disjoint triangles in $\mathbb{R}^3$ can be cut into $O(n^{7/4} + \gamma(3n) \cdot n^{1/4})$ triangular fragments such that the resulting set of fragments admits a depth order.

Proof: Define OPT to be the minimum size of a complete cut set for $\mathcal{E}$ and, for a prism $\sigma \in \mathcal{S}$, define OPT_σ to be the minimum size of a complete cut for $\mathcal{E}(\sigma)$. Then

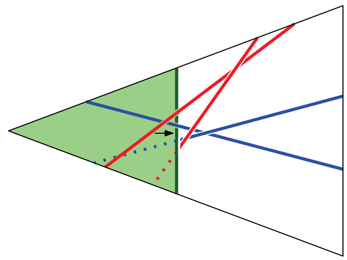

Figure 3. An example showing that one sometimes needs more cuts to eliminate all cycles from $\mathcal{T}_1(\sigma')$ than from $\mathcal{E}(\sigma')$. The set $\mathcal{T}_1(\sigma')$ consists of the green triangle, and two red and two blue segments. (The red and blue segments can easily be replaced by very thin triangles.) The set $\mathcal{E}(\sigma')$ consists of the dark green edge of the green triangle, and the red and blue segments. Note that the configuration shown in the figure is realizable.

$\sum_{\sigma\in\mathcal{S}} \text{OPT}_\sigma \leqslant \text{OPT}$. Indeed, if X_{opt} denotes a minimum-size complete cut set for $\mathcal{E}$, then $X_{\text{opt}} \cap \sigma$ must eliminate all cycles from $\mathcal{E}(\sigma)$. Since $\text{OPT} \leqslant \gamma(3n)$, the bound on the number of fragments generated by our cutting procedure is as claimed.

The procedure above cuts the triangles in $\mathcal{T}$ into constant-complexity polygonal fragments, which we can obviously cut further into triangular fragments without increasing the number of fragments asymptotically. ■

The results of Aronov and Sharir [4] thus imply that any set of n triangles can be cut into $O(n^{7/4} \operatorname{polylog} n)$ fragments such that the resulting set of fragments is acyclic. (Aronov and Sharir assume general position, but in the Section V we show this is not necessary.)

Remark 4: We use a factor $O(n^{1/4})$ more cuts than Aronov and Sharir need for the case of segments. Observe that we already generate up to $\Theta(n^{7/4})$ fragments in Step 1, since we take $r = n^{3/4}$. To reduce the total number of fragments to $O(n^{3/2} \operatorname{polylog} n)$ using our approach, we would need to set $r := \sqrt{n}$ in Step 1. In Step 2 we could then only use the set $\mathcal{V}(\sigma)$ to generate the vertical planes in $H(\sigma)$. This would lead to vertical prisms that do not have any vertex in their interior, while only using $O(n^{3/2})$ fragments so far. However, each such prism $\sigma' \subseteq \sigma$ can contain up to $\Theta(\sqrt{n})$ triangle fragments. Hence, we cannot afford to compute a cut set $X(\sigma')$ for $\mathcal{E}(\sigma')$ and cut each triangle fragment in σ' with a vertical plane containing each $q \in X(\sigma')$. One may hope that if we can eliminate all cycles from $\mathcal{E}(\sigma')$ using $|X(\sigma')|$ cuts, then we can also eliminate all cycles from $\mathcal{T}_1(\sigma')$ using $|X(\sigma')|$ cuts. Unfortunately this is not the case, as shown in Fig. 3.

In the example, there are two cycles in $\mathcal{T}_1(\sigma')$: the green triangle together with the blue segments and the green triangle with the red segments. The set $\mathcal{E}(\sigma')$ also contains two cycles. The cycles from $\mathcal{E}(\sigma')$ can be eliminated by cutting the green edge at the point indicated by the arrow. However, a single cut of the green triangle cannot eliminate both cycles from $\mathcal{T}_1(\sigma')$. Indeed, to eliminate the blue-green cycle the cut should separate (in the projection) the parts of the blue edges projecting onto the green triangle, while to eliminate the red-green cycle the cut should separate the parts of the red edges projecting onto the green triangle—but a single cut cannot do both. The example can be generalized to sets $\mathcal{T}_1(\sigma')$ of arbitrary size, so that all cuts in $\mathcal{E}(\sigma')$ can be eliminated by a single cut, while eliminating cycles from $\mathcal{T}_1(\sigma')$ requires $\Omega(|\mathcal{T}_1(\sigma')|)$ cuts. Thus a more global reasoning is needed to improve our bound.

III. Efficient algorithms to compute complete cut sets

A. The algorithm for triangles

The hierarchical cutting Ψ can be computed in $O(nr) = O(n^{7/4})$ time [14], [18], and it is easy to see that we can compute the set $\mathcal{T}_1$ within the same time bound. Constructing the 3-dimensional subdivision $\mathcal{S}$ can trivially be done in $O(n^{5/2})$ time, by checking for each of the $O(n^{3/2})$ columns and each triangle $T \in \mathcal{T}$ if T slices the column. Next we need to find the sets $\mathcal{T}_1(\sigma)$ for each prism σ in $\mathcal{S}$. The computation of the hierarchical cutting also tells us for each cell $\Delta \in \Xi_k$ which projected triangle edges intersect Δ. It remains to check, for each triangle T corresponding to such an edge, which of the $O(n)$ prisms of the column C_Δ is intersected by T. Thus, we spend $O(n)$ time for each of the $O(n^{7/4})$ triangle pieces in $\mathcal{T}_1$, so the total time to compute the sets $\mathcal{T}_1(\sigma)$ is $O(n^{11/4})$.

Next we need to compute the cut sets $X(\sigma)$. To this end we use the algorithm by Aronov *et al.* [3], which computes a complete cut set of size $O(\text{OPT}_\sigma \cdot \log \text{OPT}_\sigma \cdot \log\log \text{OPT}_\sigma)$, where OPT_σ is the minimum size of a complete cut set for $\mathcal{E}(\sigma)$. Thus $|X|$, the total size of all cut sets $X(\sigma)$ we compute, is bounded by

$$\begin{aligned} &O\left(\textstyle\sum_{\sigma\in\mathcal{S}} \text{OPT}_\sigma \cdot \log \text{OPT}_\sigma \cdot \log\log \text{OPT}_\sigma\right) \\ &\quad = O(\text{OPT} \cdot \log \text{OPT} \cdot \log\log \text{OPT}) \\ &\quad = O(n^{3/2} \operatorname{polylog} n). \end{aligned}$$

Now define $n_\sigma := |T_1(\sigma)|$. Since the algorithm of Aronov *et al.* runs in time $O(m^{4+2\omega} \log^2 m)$ for m segments, the total running time is

$$O\left(\sum_{\sigma\in\mathcal{S}} n_\sigma^{4+2\omega} \log^2 n_\sigma\right).$$

Since $n_\sigma = O(n^{1/4})$ for all σ and $\sum_{\sigma\in\mathcal{S}} n_\sigma = O(n^{7/4})$, the total time to compute the sets $X(\sigma)$ is

$$\begin{aligned} O\left(\textstyle\sum_{\sigma\in\mathcal{S}} n_\sigma^{4+2\omega} \log^2 n_\sigma\right) &= O\left(n^{3/2} \cdot (n^{1/4})^{4+2\omega} \log^2 n\right) \\ &= O(n^{5/2+\omega/2} \log^2 n). \end{aligned}$$

Finally, for each prism σ we cut all triangles in $\mathcal{T}_1(\sigma)$ by the planes in $H(\sigma)$ in a brute-force manner, in total time $O(n^{7/4} \operatorname{polylog} n)$.

The following theorem summarizes our main result.

Theorem 5: Any set $\mathcal{T}$ of n disjoint non-vertical triangles in $\mathbb{R}^3$ can be cut into $O(n^{7/4} \operatorname{polylog} n)$ triangular fragments such that the resulting set of fragments admits a depth order. The time needed to compute the cuts is $O(n^{5/2+\omega/2} \log^2 n)$, where $\omega < 2.373$ is the exponent in the running time of the best matrix-multiplication algorithm.

B. A fast algorithm for lines

The running time in Theorem 5 is better than the running time obtained by Aronov and Sharir [4] to compute a complete cut set for a set of lines in $\mathbb{R}^3$. The reason is that we apply the algorithm of Aronov *et al.* [3] locally, on a set of segments whose size is significantly smaller than n. We can use the same idea to speed up the algorithm to compute a complete cut set of size $O(n^{3/2} \operatorname{polylog} n)$ for a set L of n lines in $\mathbb{R}^3$. To this end we project L onto the xy-plane, and compute a $(1/r)$-cutting Ξ for $\overline{L}$ of size $O(r^2)$, with $r := \sqrt{n}$. We then cut each line $\ell \in L$ at the points where its projection $\overline{\ell}$ is cut by the cutting, that is, where $\overline{\ell}$ crosses the boundary of a cell Δ in Ξ. Up to this point we make only $O(nr) = O(n^{3/2})$ cuts, which does not affect the worst-case asymptotic bound on the number of cuts.

Each cell Δ of the cutting defines a vertical column C_Δ. Within each column, we apply the algorithm of Aronov *et al.* [3] to compute a cut set of size $O(\text{OPT}_\Delta \cdot \log \text{OPT}_\Delta \cdot \log\log \text{OPT}_\Delta)$, where OPT_Δ is the size of an optimal cut set inside the column. In total this gives $O(\text{OPT} \cdot \log \text{OPT} \cdot \log\log \text{OPT}) = O(n^{3/2} \operatorname{polylog} n)$ cuts in time $O(n \cdot (n^{1/2})^{4+2\omega}) = O(n^{3+\omega})$.

This leads to the following result.

Theorem 6: For any set L of n disjoint lines in $\mathbb{R}^3$, we can compute in $O(n^{3+\omega})$ time a set of $O(n^{3/2} \operatorname{polylog} n)$ cut points on the lines such that the resulting set of fragments admits a depth order, where $\omega < 2.373$ is the exponent in the running time of the best matrix-multiplication algorithm.

IV. A More Refined Bound and an Extension to Surface Patches

Let $\mathcal{T}$ be a set of disjoint surface patches in $\mathbb{R}^3$. We assume each surface patch is xy-monotone, that is, each vertical line intersects a patch in a single point or not at all, and we assume each surface patch is bounded by a constant number of bounded-degree algebraic arcs. We refer to the arcs bounding a surface patch as the *edges* of the surface patch. We assume the edges are in general position as defined by Aronov and Sharir [4], except that edges of the same patch may share endpoints. We will show how to cut the patches from $\mathcal{T}$ into fragments such that the resulting fragments admit a depth order. The total number of fragments will depend on K, the number of intersections between the projections of the edges: for any fixed $\varepsilon > 0$, we can tune our procedure so that it generates $O(n^{1+\varepsilon} + n^{1/4}K^{3/4} \operatorname{polylog} n)$ fragments. Trivially this implies that the same intersection-sensitive bound holds for triangles.

The extension of our procedure to obtain an intersection-sensitive bound for surface patches is fairly straightforward. First we observe that the analog of Proposition 1 still holds, where the base of the column can now have curved edges. In fact, the proof holds verbatim, if we allow the links of the witness cycles $\Gamma(\mathcal{C})$ that connect points a_i and b_i on the same surface patch to be curved. Now, instead of using efficient hierarchical cuttings [14], [18] we recursively generate a sequence of cuttings using the intersection-sensitive cuttings of De Berg and Schwarzkopf [12]. (This is somewhat similar to the way in which Aronov and Sharir [4] obtain an intersection-sensitive bound on the number of cuts needed to eliminate all cycles for a set of line segments in $\mathbb{R}^3$.) Below we give the details.

Let $\mathcal{E}$ denote the set of $O(n)$ edges of the surface patches in $\mathcal{T}$. A $(1/r)$-cutting for $\overline{\mathcal{E}}$ is a subdivision of $\mathbb{R}^2$ into trapezoidal cells, such that the interior of each cell intersects at most n/r edges from $\overline{E}$. Here a trapezoidal cell is a cell bounded by at most two segments that are parallel to the y-axis and at most two pieces of edges in $\overline{\mathcal{E}}$, at most one bounding it from above and at most one bounding it from below. Set $r := \min(n^{5/4}/K^{1/4}, n)$. Let ρ be a sufficiently large constant, and let k be such that $\rho^{k-1} < r \leqslant \rho^k$; the exact value of ρ depends on the desired value of ε in the final bound. We recursively construct a hierarchy $\Psi := \Xi_0, \Xi_1, \dots, \Xi_k$ of cuttings such that Ξ_i is a $(1/\rho^i)$-cutting for $\overline{\mathcal{E}}$, as follows. The initial cutting Ξ_0 is the entire plane $\mathbb{R}^2$. To construct Ξ_i we take each cell Δ of Ξ_{i-1} and we construct a $(1/\rho)$-cutting for the set $\overline{\mathcal{E}}_\Delta := \{\overline{e} \cap \Delta : \overline{e} \in \overline{\mathcal{E}}\}$. De Berg and Schwarzkopf [12] have shown that there is such a cutting consisting of $O(\rho + K_\Delta \rho^2/n_\Delta^2)$ cells, where $n_\Delta := |\overline{\mathcal{E}}_\Delta|$ and K_Δ is the number of intersections inside Δ. One easily shows by induction on i that for each cell Δ in Ξ_{i-1} we have[2] $n_\Delta \leqslant n/\rho^{i-1}$. Hence, by combining the cuttings Ξ_Δ over all $\Delta \in \Xi_{i-1}$ we obtain a $(1/\rho^i)$-cutting Ξ_i.

Let $|\Xi_i|$ be the number of cells in Ξ_i. Then $|\Xi_0| = 1$ and, for a suitable constant D—the constant D depends on the degree of the edges and follows from the construction of De Berg and Schwarzkopf [12]—we have

$$\begin{aligned}
|\Xi_i| &\leqslant \textstyle\sum_{\Delta \in \Xi_{i-1}} D(\rho + K_\Delta \rho^2/n_\Delta^2) \\
&\leqslant D\rho \cdot |\Xi_{i-1}| + DK\rho^{2i}/n^2 \\
&\qquad \text{(since } \textstyle\sum_{\Delta \in \Xi_{i-1}} K_\Delta \leqslant K \text{ and } n_\Delta \leqslant n/\rho^{i-1}\text{)} \\
&\leqslant D^i\rho^i + \tfrac{DK}{n^2} \textstyle\sum_{j=0}^{i} D^j \rho^{2i-j} \\
&\leqslant D^i\rho^i + \tfrac{DK}{n^2} \cdot 2\rho^{2i} \quad \text{(assuming } \rho > 2D\text{).}
\end{aligned}$$

[2]Strictly speaking this is not accurate, as n denotes the number of patches and not the number of edges. To avoid cluttering the notation we allow ourselves this slight abuse of notation.

Now we can proceed exactly as before. Thus we first traverse the hierarchy Ψ with each patch $T \in \mathcal{T}$, associating T to nodes u such that $\Delta_u \subseteq \overline{T}$ and $\Delta_{\mathrm{parent}(u)} \not\subseteq \overline{T}$, and to the leaves that we reach. This partitions T into a number of fragments. The resulting set $\mathcal{T}_1$ of fragments generated over all triangles $T \in \mathcal{T}$ has total size

$$\begin{aligned} |\mathcal{T}_1| &= O\left(\textstyle\sum_{i=0}^{k-1}\sum_{\Delta\in\Xi_i} n/\rho^i\right) \qquad (2)\\ &= O\left(\textstyle\sum_{i=0}^{k-1}\left(D^i\rho^i + \frac{DK}{n^2}\rho^{2i}\right)\cdot(n/\rho^i)\right)\\ &= O\left(n\textstyle\sum_{i=0}^{k-1} D^i + \frac{DK}{n}\sum_{i=0}^{k-1}\rho^i\right)\\ &= O\left(nD^k + DKr/n\right)\end{aligned}$$

If we now set $\rho := D^{1/\varepsilon}$ then $D^k = \rho^{k\varepsilon} = O(r^\varepsilon) = O(n^\varepsilon)$, and so $|\mathcal{T}_1| = O(n^{1+\varepsilon} + Kr/n)$. We then decompose $\mathbb{R}^3$ into a subdivision $\mathcal{S}$ consisting of prisms σ that each intersect at most n/r surface patches, take a minimum-size complete cut set $X(\sigma)$ for the edges inside each prism σ, and generate a set $H(S)$ of cutting planes through the points in $X(\sigma) \cup \mathcal{V}(\sigma)$. Here $\mathcal{V}(\sigma)$ is, as before, the set of vertices of the surface patches in the interior of σ. Since Aronov and Sharir [4] proved that any set of n bounded-degree algebraic arcs in general position[3] admits a complete cut set of size $O(n + (nK)^{1/2}\operatorname{polylog} n)$—the constant of proportionality and the exponent of the polylogarithmic factor depend on the degree of the arcs—we have

$$\sum_{\sigma\in\mathcal{S}} |X(\sigma)| = O\left(n + (nK)^{1/2}\operatorname{polylog} n\right).$$

Hence, the number of extra fragments created in Step 2 is

$$\begin{aligned} &O(n/r)\cdot\left(\textstyle\sum_{\sigma\in\mathcal{S}}(|\mathcal{V}(\sigma)| + |X(\sigma)|)\right)\\ &\quad = O\left(n^2/r + (n^{3/2}K^{1/2}/r)\operatorname{polylog} n\right).\end{aligned}$$

By picking $r := \min(n^{5/4}/K^{1/4}, n)$ our final bound on the number of fragments becomes $O(n^{1+\varepsilon} + n^{1/4}K^{3/4}\operatorname{polylog} n)$.

Theorem 7: Let $\mathcal{T}$ be a set of n disjoint xy-monotone surface patches in $\mathbb{R}^3$, each bounded by a constant number of constant-degree algebraic arcs in general position. Then for any fixed $\varepsilon > 0$ we can cut $\mathcal{T}$ into $O(n^{1+\varepsilon} + n^{1/4}K^{3/4}\operatorname{polylog} n)$ fragments that admit a depth order, where K is the number of intersections between the projections of the edges of the surface patches in $\mathcal{T}$. The constant of proportionality and the exponent of the polylogarithmic factor depend on the degree of the edges. The expected time needed to compute the cuts is $O\left(n^{1+\varepsilon} + K^{(3+\omega)/2+\varepsilon}/n^{(1+\omega)/2}\right)$, where $\omega < 2.373$ is the exponent in the running time of the best matrix-multiplication algorithm.

Proof: The bound on the number of fragments follows from the discussion above. To prove the time bound we first note that an intersection-sensitive cutting of size $O(\rho + K_\Delta\rho^2/n_\Delta^2)$ can be computed in expected time $O(n_\Delta \log\rho + K_\Delta\rho/n_\Delta)$ [12]. Hence, constructing the hierarchy takes expected time

$$\begin{aligned} &O\left(\textstyle\sum_{i=0}^{k-1}\sum_{\Delta\in\Xi_i}\left(\frac{n}{\rho^i}\log\rho + K_\Delta\frac{\rho}{(n/\rho^i)}\right)\right)\\ &\quad = O\left(\textstyle\sum_{i=0}^{k-1}\sum_{\Delta\in\Xi_i}\frac{n}{\rho^i}\log\rho\right) + O\left(\textstyle\sum_{i=0}^{k-1} K\frac{\rho^{i-1}}{n}\right).\end{aligned}$$

The first term is the same as in Equation (2) except for the extra $\log\rho$-factor (which is a constant), so this term is still bounded by $O(n^{1+\varepsilon} + Kr/n)$. Since $\rho^{k-1} \leqslant r$, the second term is bounded by $O(Kr/n)$, which is dominated by the first term. Thus the total expected time to compute the set $\mathcal{T}_1$ is $O(n^{1+\varepsilon} + Kr/n)$.

In the second stage of the algorithm we use the algorithm of Aronov *et al.* [3] on the set $\mathcal{E}(\sigma)$ of edge fragments inside each prism $\sigma \in \mathcal{S}$. Aronov *et al.* only explicitly state their result for line segments, but it is easily checked that it works for curves as well; the fact that, for example, there can already be cyclic overlap between a pair of curves has no influence on the algorithm's approximation factor or running time. Indeed, the crucial property still holds that cut points can be ordered linearly along a curve, and this is sufficient for the algorithm to work. Thus the time needed to compute all cut sets $X(\sigma)$ is

$$O\left(\sum_{\sigma\in\mathcal{S}} n_\sigma^{4+2\omega}\log^2 n_\sigma\right),$$

where n_σ is the number of edges inside σ. Since $n_\sigma = O(n/r)$ and $\sum_{\sigma\in\mathcal{S}} n_\sigma = O(|\mathcal{T}_1|) = O(n^{1+\varepsilon} + Kr/n)$, computing the cut sets takes

$$O\left(\frac{n^{4+2\omega+\varepsilon}}{r^{3+2\omega}} + K\left(\frac{n}{r}\right)^{2+2\omega+\varepsilon}\right)$$

time, for a slightly larger ε than before. Because we picked $r := \min(n^{5/4}/K^{1/4}, n)$, the time to compute the cut sets is

$$\begin{aligned} &O\left(n^{1+\varepsilon} + n^{\frac{1}{4}-\omega/2+\varepsilon}\cdot K^{\frac{3}{4}+\omega/2} + \frac{K^{\frac{3}{2}+\omega/2}+\varepsilon}{n^{\frac{1}{2}+\omega/2}}\right)\\ &\quad = O\left(n^{1+\varepsilon} + K^{\frac{3}{2}+\omega/2+\varepsilon}/n^{\frac{1}{2}+\omega/2}\right),\end{aligned}$$

which dominates the time for the first stage. Finally, cutting the patches inside each prism σ in a brute-force manner takes time linear in the maximum number of fragments we generate, namely $O(n^{1+\varepsilon} + n^{1/4}K^{3/4}\operatorname{polylog} n)$. Thus the total expected time is

$$O\left(n^{1+\varepsilon} + K^{\frac{3}{2}+\omega/2+\varepsilon}/n^{\frac{1}{2}+\omega/2}\right).$$

∎

[3]We assumed the edges are in general position, but edges of the same patch may share endpoints. However, there are no endpoints in the interior of σ, and so the only degeneracy that can happen is if two edges of the same patch share an endpoint that lies on the boundary of σ. In this case we can slightly shorten and perturb the edges to remove this degeneracy as well.

Observe that for $K = n^2$ the bound in the theorem above is essentially the same bound as in Theorem 5.

V. Dealing with degeneracies

We first show how to deal with degeneracies when eliminating cycles from a set of segments (or lines) in $\mathbb{R}^3$, and then we argue that the method for triangles presented in the main text does not need any non-degeneracy assumptions either. We do not deal with removing the non-degeneracy assumptions for the case of surface patches.

A. Degeneracies among segments

Let $S = \{s_1, \ldots, s_n\}$ be a set of disjoint segments in $\mathbb{R}^3$. (Even though we allow degeneracies, we do not allow the segments in S to intersect or touch, since then the problem is not well-defined. If the segments are defined to be relatively open, then we can also allow an endpoint of one segment to coincide with an endpoint of, or lie in the interior of, another segment.) We can assume without loss of generality that S does not contain vertical segments, since eliminating all cycles from the non-vertical segments in S also eliminates all cycles when we include the vertical segments. Aronov and Sharir [4] make the following non-degeneracy assumptions:

(i) no endpoint of one segment projects onto any other segment;
(ii) no three segments are concurrent (that is, pass through a common point) in the projection;
(iii) no two segments in S are parallel.

The main difficulty arises from type (iii) degeneracies where parallel segments overlap in the projection. The problem is that a small perturbation will reduce the intersection in the projection to a single point, and cutting one of the segments at the intersection is effective for the perturbed segments but not necessarily for the original segments. Next we describe how we handle this and how to deal with the other degeneracies as well.

First we slightly extend each segment in S—segments that are relatively open would be slightly shortened—to get rid of degeneracies of type (i), and we slightly translate each segment to make sure no two segments intersect in more than a single point in the projection. (The translations are not necessary, but they simplify the following description and bring out more clearly how the $\prec$-relations between parallel segments are treated.) Next, we slightly perturb each segment such that all degeneracies disappear and any two non-parallel segments whose projections intersect before the perturbation still do so after the perturbation. This gets rid of degeneracies of types (ii) and (iii). Let s'_i denote the segment s_i after the perturbation, and define $S' := \{s'_1, \ldots, s'_n\}$. The set S' has the following properties:

- for any two non-parallel segments $s_i, s_j \in S$ we have $s_i \prec s_j$ if and only if $s'_i \prec s'_j$;
- the order of intersections along segments in the projection is preserved in the following sense: if $\overline{s'_i} \cap \overline{s'_j}$ lies before $\overline{s'_i} \cap \overline{s'_k}$ along $\overline{s'_i}$ as seen from a given endpoint of $\overline{s'_i}$, then $\overline{s_i} \cap \overline{s_j}$ does not lie behind $\overline{s_i} \cap \overline{s_k}$ along $\overline{s_i}$ as seen from the corresponding endpoint of $\overline{s_i}$;
- if s_i and s_j are parallel then $\overline{s'_i}$ and $\overline{s'_j}$ do not intersect.

We will show how to obtain a complete cut set for S from a complete cut set X' for S'. The cut set for S will consist of a cut set X that is derived from X' plus a set Y of $O(n \log n)$ additional cuts, as explained next.

- Let $q' \in X'$ be a cut point on a segment $s'_i \in S'$. Let $s'_j \in S'$ be the segment such that $\overline{s'_i} \cap \overline{s'_j}$ is the intersection point on $\overline{s'_i}$ closest to $\overline{q'}$, with ties broken arbitrarily. We can assume that s'_j exists, since if $\overline{s'_i}$ does not intersect any projected segment then the cut point q is useless and can be ignored. Now we put into X the point $q \in s_i$ such that $\overline{q} = \overline{s_i} \cap \overline{s_j}$. (It can happen that several cut points along s'_i generate the same cut point along s_i. Obviously we need to insert only one of them into X.) The crucial property of the cut point $q \in X$ generated for $q' \in X'$ is the following:
  - if $\overline{q'}$ coincides with a certain intersection along $\overline{s'_i}$ then $\overline{q}$ coincides with the corresponding intersection along $\overline{s_i}$;
  - if $\overline{q'}$ separates two intersections along $\overline{s'_i}$ then $\overline{q}$ separates the corresponding intersections along $\overline{s_i}$ or it coincides with at least one of them.

  By treating all cut points in X' in this manner, we obtain the set X.
- The set Y deals with parallel segments in S whose projections overlap. It is defined as follows. Let $S(X)$ be the set of fragments resulting from cutting the segments in S at the cut points in X. Partition $S(X)$ into subsets $S_\ell(X)$ such that $S_\ell(X)$ contains all fragments from $S(X)$ projecting onto the same line ℓ. Consider such a subset $S_\ell(X)$ and assume without loss of generality that ℓ is the x-axis. Construct a segment tree [9] for the projections of the fragments in $S_\ell(X)$. Each projected fragment $\overline{f}$ is stored at $O(\log |S_\ell(X)|) = O(\log n)$ nodes of the segment tree, which induces a subdivision of $\overline{f}$ into $O(\log n)$ intervals. We put into Y the $O(\log n)$ points on f whose projections define these intervals. The crucial property of segment trees that we will need is the following:
  - Let I_v denote the interval corresponding to a node v. Then for any two nodes v, w we either have $I_v \subseteq I_w$ (when v is a descendent of w), or we have $I_v \supseteq I_w$ (when v is an ancestor of w), or otherwise the interiors of I_v and I_w are disjoint. Hence, a similar property holds for the projections of the sub-fragments resulting from cutting the fragments in $S_\ell(X)$ as explained above.

  Doing this for all fragments $s_i \in S_\ell(X)$ and for all subsets $S_\ell(X)$ gives us the extra cut set Y.

Lemma 8: *The set* $X \cup Y$ *is a complete cut set for* S.

Proof: Let F denote the set of fragments resulting from cutting the segments in S at the points in $X \cup Y$, and suppose for a contradiction that F still contains a cycle. Let $\mathcal{C} := f_0 \prec f_1 \prec \cdots \prec f_{k-1} \prec f_0$ be a minimal cycle in F, and let $s_i \in S$ be the segment containing f_i.

As explained above, the cut points in Y guarantee that for any two parallel fragments in F whose projections overlap, one is contained in the other in the projection. This implies that two consecutive fragments f_i, f_{i+1} in $\mathcal{C}$ cannot be parallel: if they were, then $\overline{f_i} \subseteq \overline{f_{i+1}}$ (or vice versa) which contradicts that $\mathcal{C}$ is minimal. Hence, any two consecutive fragments are non-parallel. Now consider the witness curve $\Gamma(\mathcal{C})$ for $\mathcal{C}$. Since consecutive fragments in $\mathcal{C}$ are non-parallel, $\Gamma(\mathcal{C})$ is unique. Let Γ' be the corresponding curve for S', that is, Γ' visits the segments $s'_0, s'_1, \ldots, s'_{k-1}, s'_0$ from S' in the given order—recall that $f_i \subseteq s_i$ and that s'_i is the perturbed segment s_i—and it steps from s'_i to s'_{i+1} using vertical connections. Since X' is a complete cut set for S', there must be a link of Γ', say on segment s'_i, that contains a cut point $q' \in X'$. In other words, $\overline{q'}$ separates $\overline{s'_{i-1}} \cap \overline{s'_i}$ from $\overline{s'_i} \cap \overline{s'_{i+1}}$, or it coincides with one of these points. But then the cut point $q \in X$ corresponding to q' must separate $\overline{s_{i-1}} \cap \overline{s_i}$ from $\overline{s_i} \cap \overline{s_{i+1}}$ or coincide with one of these points, thus cutting the witness curve $\Gamma(\mathcal{C})$—a contradiction. ■

Theorem 9: *Suppose any non-degenerate set of* n *disjoint segments can be cut into* $\gamma(n)$ *fragments in* $T(n)$ *time such that the resulting set of fragments admits a depth order. Then any set of* n *disjoint segments can be cut into* $O(\gamma(n) \log n)$ *fragments in* $T(n) + O(n^2)$ *time such that the resulting set of fragments admits a depth order.*

Proof: The bound on the number of fragments immediately follows from the discussion above. The overhead term in the running time is caused by the computation of the perturbed set S', which can be done in $O(n^2)$ time if we compute the full arrangement in the projection. ■

B. Degeneracies among triangles

Recall that the cuts we make on the triangles are induced by vertical planes, and that a triangle becomes open where it is cut. When a triangle is completely contained in the cutting plane, however, it is not well defined what happens. One option is to say that the triangle completely disappears; another option it to say that the triangle is not cut at all. Since vertical triangles can be ignored in the Painter's Algorithm, we will simply assume that no triangle in $\mathcal{T}$ is vertical. However, we can still have other degeneracies, such as edges of different triangles being parallel or triples of projected edges being concurrent. Fortunately, the fact that we do not need non-degeneracy assumptions for segments immediately implies that we can handle such cases. Indeed, degeneracies are not a problem for the hierarchical cuttings we use in Step 1 of our procedure, and in Step 2 we only assumed non-degeneracy when computing the cut set $X(\sigma)$ for the edge set $\mathcal{E}(\sigma)$—and Theorem 9 implies we can get rid of the non-degeneracy assumptions in this step. Note that the $O(n^2)$ overhead term in Theorem 9 is subsumed by the time needed to apply the algorithm of Aronov *et al.* [3].

VI. Concluding remarks

We proved that any set of n disjoint triangles in $\mathbb{R}^3$ can be cut into $O(n^{7/4} \operatorname{polylog} n)$ triangular fragments that admit a depth order, thus providing the first subquadratic bound for this important setting of the problem. We also proved a refined bound that depends on the number of intersections of the triangle edges in the projection, and generalized the result to xy-monotone surface patches. The main open problem is to tighten the gap between our bound and the $\Omega(n^{3/2})$ lower bound on the worst-case number of fragments needed: is it possible to cut any set of triangles into roughly $\Omega(n^{3/2})$ triangular fragments that admit a depth order, or is this only possible by using curved cuts? One would expect the former, but curved cuts seem unavoidable in the approach of Aronov, Miller and Sharir [6] and it seems very hard to push our approach to obtain an $o(n^{7/4})$ bound.

References

[1] P.K. Agarwal, E.F. Grove, T.M. Murali, and J.S. Vitter. Binary space partitions for fat rectangles. *SIAM J. Comput.* 29:1422–1448 (2000).

[2] P.K. Agarwal, M.J. Katz, and M. Sharir. Computing depth orders for fat objects and related problems. *Comput. Geom. Theory Appl.* 5:187–206 (1995).

[3] B. Aronov, M. de Berg, C. Gray, and E. Mumford. Cutting cycles of rods in space: Hardness results and approximation algorithms. In *Proc. 19th ACM-SIAM Symp. Discr. Alg. (SODA)*, pages 1241–1248, 2008.

[4] B. Aronov and M. Sharir. Almost tight bounds for eliminating depth cycles in three dimensions. In *Proc. 48th ACM Symp. Theory Comp. (STOC)*, pages 1–8, 2016.

[5] B. Aronov, V. Koltun, and M. Sharir. Cutting triangular cycles of lines in space. *Discr. Comput. Geom.* 33:231–247 (2005).

[6] B. Aronov, E.Y. Miller, and M. Sharir. Eliminating depth cycles among triangles in three dimensions. In *Proc. 28th ACM-SIAM Symp. Discr. Alg. (SODA)*, pages 2476–2494, 2017.

[7] M. de Berg. *Ray Shooting, Depth Orders and Hidden Surface Removal*. LNCS 703, Springer-Verlag, 1993.

[8] M. de Berg. Linear size binary space partitions for uncluttered scenes. *Algorithmica* 28:353–366 (2000).

[9] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Computational Geometry: Algorithms and Applications.* Springer, 2008.

[10] M. de Berg and C. Gray. Vertical ray shooting and computing depth orders for fat objects. *SIAM J. Comput.* 38:257–275 (2008).

[11] M. de Berg, M. Overmars, and O. Schwarzkopf. Computing and verifying depth orders. *SIAM J. Comput.* 23:437–446 (1994).

[12] M. de Berg and O. Schwarzkopf. Cuttings and applications. *Int. J. Comput. Geom. Appl.* 5:43–55 (1995).

[13] B. Chazelle. Convex partitions of polyhedra: A lower bound and worst-case optimal algorithm. *SIAM J. Comput.* 13:488–507 (1984).

[14] B. Chazelle. Cutting hyperplanes for divide and conquer. *Discr. Comput. Geom.* 9:145–158 (1993).

[15] B. Chazelle, H. Edelsbrunner, L.J. Guibas, R. Pollack, R. Seidel, M. Sharir, and J. Snoeyink. Counting and cutting cycles of lines and rods in space. *Comput. Geom. Theory Appl.* 1:305–323 (1992). (A preliminary version appeared in In *Proc. 31st IEEE Symp. Found. Comput. Sci. (FOCS)*, pages 242–251, 1991.)

[16] L. Guth. Polynomial partitioning for a set of varieties. *Math. Proc. Cambridge Phil. Soc.* 159:459–469 (2015).

[17] S. Har-Peled and M. Sharir. Online point location in planar arrangements and its applications. *Discr. Comput. Geom.* 26: 19–40 (2001).

[18] J. Matoušek. Range searching with efficient hierarchical cuttings. *Discr. Comput. Geom.* 10: 157–182 (1993).

[19] M. S. Paterson and F. F. Yao. Efficient binary space partitions for hidden-surface removal and solid modeling. *Discr. Comput. Geom.* 5:485–503 (1990).

[20] M. S. Paterson and F. F. Yao. Optimal binary space partitions for orthogonal objects. *J. Alg.* 13:99–113 (1992).

[21] A. Solan. Cutting cycles of rods in space. In *Proc. 14th Int. Symp. Comput. Geom. (SoCG)*, pages 135–142, 1998.

[22] C.D. Tóth. Binary space partitions for axis-aligned fat rectangles. *SIAM J. Comput.* 38: 429–447 (2008).

[23] R.H. Wilson and J.-C. Latombe. Geometric reasoning about mechanical assembly. *Artificial Intelligence* 71:371–396 (1994).