Eliminating Depth Cycles Among Triangles in Three Dimensions

Boris Aronov¹ · Edward Y. Miller² · Micha Sharir³

Received: 7 March 2019 / Revised: 7 April 2020 / Accepted: 28 May 2020 / Published online: 14 July 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

The vertical depth relation among $n$ pairwise openly disjoint triangles in 3-space may contain cycles. We show that, for any $\varepsilon > 0$, the triangles can be cut into $O(n^3/2+\varepsilon)$ connected semialgebraic pieces, whose description complexity depends only on the choice of $\varepsilon$, such that the depth relation among these pieces is now a proper partial order. This bound is nearly tight in the worst case. The pieces can be constructed efficiently. This work extends the recent study by two of the authors (Discrete Comput. Geom. 59(3), 725–741 (2018)) on eliminating depth cycles among lines in 3-space. Our approach is again algebraic, and makes use of a recent variant of the polynomial partitioning technique, due to Guth (Math. Proc. Camb. Philos. Soc. 159(3), 459–469 (2015)), which leads to a recursive algorithm for cutting the triangles. In contrast to the case of lines, our analysis here is considerably more involved, due to the two-dimensional nature of the objects being cut, so additional tools, from topology and algebra, need to be brought to bear. Our result makes significant progress towards resolving a decades-old open problem in computational geometry, motivated by hidden-surface removal in computer graphics. In addition, we generalize our bound to well-behaved patches of two-dimensional algebraic surfaces of constant degree.

To Ricky, a mathematician, an inspiration, a friend, who brought us all together

Work on this paper by B.A. has been partially supported by NSF Grants CCF-11-17336, CCF-12-18791, and CCF-15-40656, and by BSF grant 2014/170. Work by M.S. has been supported by Grant 2012/229 from the U.S.–Israel Binational Science Foundation, by Grants 892/13 and 260/18 from the Israel Science Foundation, by the Israeli Centers for Research Excellence (I-CORE) program (center no. 4/11), by the Hermann Minkowski–MINERVA Center for Geometry at Tel Aviv University, and by Grant G-1367-407.6/2016 from the German–Israeli Foundation for Scientific Research and Development. An earlier version of this work appeared in SODA’17 [6].

Extended author information available on the last page of the article
Keywords  Depth order · Depth cycles · Cycle elimination · Painter’s algorithm · 
Algebraic methods in combinatorial geometry · Polynomial partitioning

Mathematics Subject Classification  68U05 · 52C99 · 52C35

We are honored to have been invited to contribute this paper to this special issue of *Discrete & Computational Geometry* dedicated to the memory of Ricky Pollack. Ricky had been a great friend of ours—in fact a friend of the entire communities of discrete and computational geometry. Together with his long-time associate, Eli Goodman, they have fostered and nurtured both communities, bringing them together in a series of major conferences, in running the extremely influential geometry seminar at the Courant Institute of Mathematical Sciences at NYU, and of course in founding and running this cornerstone journal of the area for many years.

But beyond all of this Ricky was an emotional loving friend, a person who cared about the people around him, a lively, energetic, intellectually curious human being. We all miss him deeply, and offer this paper as a farewell present to him.

The topic of this paper was close to Ricky’s interests. He wrote in 1993, together with János Pach and Emo Welzl, a paper on weaving patterns of lines and line segments in space, which studies the patterns in which a finite set of lines or line segments in 3-space can pass above and below each other. Results of this type have been instrumental in the earlier studies of the cycle-breaking problem, before the new algebraic machinery was brought into the game. In this sense, the present paper (almost) closes the lid on the topic initiated by Ricky and others many years ago.

Again, we are very glad to be part of this project. Ricky’s memory is in our hearts, and we miss him sorely.

1 Introduction

The Problem  Let $T$ be a collection of $n$ non-vertical pairwise disjoint triangles in $\mathbb{R}^3$. We treat the triangles as relatively open and allow their boundaries to intersect or overlap. For any pair $\Delta, \Delta'$ of triangles in $T$, we say that $\Delta$ passes *above* $\Delta'$ (or $\Delta'$ passes *below* $\Delta$) if there exists a vertical line that meets both $\Delta$ and $\Delta'$ so that it intersects $\Delta$ at a point that lies higher than its intersection with $\Delta'$; this property is clearly independent of the choice of the vertical line meeting both triangles. We denote this partial relation by $\Delta' \prec \Delta$ or $\Delta \succ \Delta'$. The relation $\prec$ in general may contain cycles of the form $\Delta_1 \prec \Delta_2 \prec \cdots \prec \Delta_k \prec \Delta_1$. We call this a $k$-cycle, and refer to $k$ as the length of the cycle, see Fig. 1.

The problem of cycle elimination is to cut the triangles of $T$ into a finite number of connected pieces, also considered as relatively open, each being semialgebraic (see below for precise definitions), so that the suitably extended depth relation among the new pieces is acyclic. When no cycles remain we call the resulting depth relation a depth order.

The simpler case, with triangles replaced by lines or line segments, has been addressed in a number of papers over the last 30 years, culminating in the work by two of the authors [7]. We refer the reader to [7] and to the recent work of de Berg [10] for
a detailed review of the problem history and related work. Note that eliminating cycles in a set of triangles adds, literally, a new dimension to the problem: whereas lines, segments, or curves need to be cut at a discrete set of points, triangles have to be cut into pieces along curves, which makes the analysis considerably more involved. We also observe that the binary space partition (BSP) technique of Paterson and Yao [21] constructs a depth order by cutting the triangles into $O(n^2)$ pieces, but, as in the case of lines, we would like to use fewer cuts, ideally close to the lower bound of $\Omega(n^{3/2})$, which is an immediate extension of a similar lower bound in [13] for the case of lines.

A long-standing conjecture for the case of lines, open since at least 1990, has been that one can indeed always construct a depth order by cutting the lines into a subquadratic number of pieces (see the conference version of [13], for example; although certain that the question has been asked earlier, we have been unable to locate a published reference). Refer to [8, Chap. 9] for a summary of the state of affairs at the beginning of the 1990s. In the previous work [7] we have shown that $O(n^{3/2} \text{polylog } n)$ cuts, and thus pieces produced, suffice to eliminate all cycles among $n$ lines in space. This settles the above conjecture, and almost attains the lower bound $\Omega(n^{3/2})$ just noted. In this paper we obtain a similar, albeit slightly weaker, bound for the case of triangles, almost settling this conjecture also for the case of triangles, in a strong, almost worst-case tight manner, except for the fact that our cuts are not by straight segments.

**Background** The main original motivation for studying this problem comes from hidden surface removal in computer graphics, as described, for example, in Aronov et al. [5] or de Berg [10] (for a discussion of applications of depth orders in computational geometry see, e.g., [8,10] and references therein). Briefly, a conceptually simple technique for rendering a scene in computer graphics is the so-called Painter’s Algorithm, which places the objects from the scene on the screen in a back-to-front manner, painting each new object over the portions of earlier objects that it hides. For this, though, one needs an acyclic depth relation among the objects with respect to the viewing point (which we assume hereafter, without loss of generality, to lie at $z = +\infty$), as the algorithm would fail if applied directly to the triangles of Fig. 1.

1 A significant feature of the BSP technique is that the cuts are made by straight lines and therefore the resulting pieces can be taken to be triangular, whereas this is not the case in our construction. Moreover, the resulting pieces have no depth cycles with respect to any viewing direction, and also for any perspective view.
say. In presence of depth cycles, one would like to cut the objects into a small number of pieces, so as to eliminate all cycles, i.e., have an acyclic depth relation among the resulting pieces. Then one can apply Painter’s Algorithm (or any other algorithm that requires a depth order) to the collection of pieces thus obtained. For efficiency, we would like to minimize the number of pieces.

Assuming that the input objects are all given as triangulated polyhedral approximations, as is the case in many practical applications, we face exactly the problem addressed in this paper.

In contrast with the progress on the case of lines in [7], mentioned above, the case of triangles has barely been touched until fairly recently. The aforementioned BSP technique in [21] handles arbitrary disjoint triangles, but produces a quadratic number of pieces, in the worst case. Several subsequent refinements establish subquadratic bounds for special classes of objects in three dimensions, such as axis-parallel two-dimensional rectangles of bounded aspect ratio [2,26], or so-called uncluttered scenes [9]; see [15,25,26] for surveys of the BSP literature.

**Recent Progress** Two algorithmic developments took place fairly recently (after the original appearance of an earlier version of this work in [6]). Agarwal et al. [1] presented an efficient randomized algorithm for constructing the polynomial partitioning for lines; the existence of such a partitioning was promised by Guth [17], see also Guth and Katz [18]. Such a partitioning is a major ingredient of the analysis in this paper, as well as in our earlier study of the case of lines [7] (see Propositions 2.1 and 2.2). The result in [1] completes the puzzle, and yields, together with the analysis in [7] and current work, an algorithm that computes, in near $O(n^{3/2})$ expected time, a near-worst-case optimal partitioning of $n$ triangles or $n$ lines into about $O(n^{3/2})$ pieces with acyclic depth order. See also the work of Aronov et al. [4] for a related partition sufficient for the case of lines. Further details are discussed in Sect. 5.

Ideally, we would like the curves partitioning the triangles to be straight and the pieces to be triangular, as yielded by the BSP technique [21]. In a more recent work of de Berg [10], which is based on our earlier work for lines [7], the triangles are indeed cut by straight segments, but it produces a larger number of pieces, with a bound close to $O(n^{7/4})$. (His algorithm for achieving the cuts runs in time $O(n^{3.69})$, in contrast to the algorithm mentioned above, with only $O(n^{3/2+\varepsilon})$ expected time.) Unfortunately, our argument in its current form cannot achieve this straightness property, even in principle. How to efficiently obtain a subquadratic number of triangular pieces with a proper depth order, in subquadratic time, with bounds closer to $O(n^{3/2})$ or just better than de Berg’s bound of about $O(n^{7/4})$, is a fascinating remaining open problem.

**Our Contribution** In this paper we partially settle the problem for the case of triangles, and show that, for any prespecified $\varepsilon > 0$, all cycles in the depth relation in a set of $n$ pairwise disjoint, relatively open triangles in $\mathbb{R}^3$ can be eliminated by cutting the triangles into $O(n^{3/2+\varepsilon})$ pieces, each a pseudo-trapezoid (see below for definitions) bounded by at most two line segments and at most two algebraic arcs of degree $O(\sqrt{\frac{1}{\varepsilon}})$. Using the machinery of Agarwal et al. [1], the method is constructive and takes expected time $O(n^{3/2+\varepsilon})$. See Theorem 4.7. Moreover, a straightforward variant of the above argument yields $n^{3/2} \cdot 2^{O(\sqrt{\log n})}$ trapezoids whose bounding algebraic curves have degree $2^{O(\sqrt{\log n})}$. As already noted, our bound on the number of pieces is close
Fig. 2 Δ1 slices the open connected region σ. Δ2 pierces σ. Δ3 also pierces σ, even though only one connected component of Δ3 ∩ σ meets the boundary of Δ3.

to the best possible in the worst case, as Ω(n^{3/2}) pieces are sometimes necessary. In fact, our results, with only rather minor modifications, apply to a more general class of objects in three dimensions, namely, arbitrary patches of algebraic surfaces, provided that the surfaces are \(x\)-\(y\)-monotone, have bounded degree, and each patch is cut out of such a surface by a constant number of bounded-degree algebraic curves. For the precise statement, see Theorem 5.2 in Sect. 5.

Before proceeding, we need a technical definition. Let \(\sigma\) be an open connected set in \(\mathbb{R}^3\). We say that a triangle \(\Delta\) intersecting \(\sigma\) pierces \(\sigma\) if the boundary of \(\Delta\) meets \(\sigma\), and that \(\Delta\) slices \(\sigma\) otherwise, i.e., if only the interior of \(\Delta\) meets \(\sigma\); see Fig. 2 for an illustration.

The proof of our bound follows the high-level approach in the previous analysis for the case of lines [7], which uses the polynomial partitioning technique of Guth [17]. Roughly speaking, this technique spreads the edges of the triangles more or less evenly among the cells of the partition, which in turn provides a recursive divide-and-conquer mechanism for performing the cuts and analyzing their number. However, the fact that we are dealing here with two-dimensional triangles, rather than with one-dimensional lines (or segments, or arcs), raises substantial technical problems that need to be overcome.

One significant issue that arises (and has already been discussed) is that, in contrast with the case of lines, where the cuts are made at a discrete set of points, here we need to cut the triangles into two-dimensional regions by algebraic curves.

Another issue involves controlling the recursive mechanism, so as to ensure that not too many triangles are passed to a recursive subproblem (each within some cell of the partition). As alluded to above, we can control the number of triangles that have an edge crossing a cell, i.e., the triangles piercing the cell, since the partition is based on the triangle edges, but we do not have a good bound on the number of triangles that slice through a cell. One therefore needs to prune away the triangles that cross a cell in this slicing manner, in order to obtain a recurrence relationship similar to the

---

2 See [7] for details. Roughly, the lower bound results from a reduction from a construction of \(\Omega(n^{3/2})\) joints in a collection of \(n\) lines in \(\mathbb{R}^3\).
Fig. 3 A donut-shaped cell of the partition is sliced by several triangles in a pinwheel fashion, thereby forming a depth cycle. Our analysis aims to get rid of such “slicing cycles,” as the number of triangles slicing a cell may be too large. Only four intersections of the cell with these triangles are shown, out of many.

one in [7] for the case of lines, thereby achieving the desired near-optimal (close to $O(n^{3/2})$) bound for the overall number of cuts.

Yet another difficulty, to which an earlier version of this work [6] devoted significant effort, is in handling the fact that cells arising from a polynomial partitioning may have non-trivial topology. Without additional care, we could have, say, a donut-shaped cell containing a cycle formed solely from triangles that slice the cell; see Fig. 3 and [6].

A considerable portion of the analysis in [6] was devoted to eliminating cycles that involve slicing triangles, by a controlled additional subdivision of the triangles. In the current technique, we use what we believe to be a cleaner approach, in which we subdivide the triangles according to their interaction with the vertical decomposition of the cell (see Sect. 4 for details). This guarantees that any cycle in the depth relation within the cell, that has not yet been eliminated, has to be confined to a single subcell of the vertical decomposition (which, by construction, has trivial topology), or to its 2-skeleton, a situation that is easy to handle separately. Using topological arguments (see Lemma 2.4), we show that such a surviving cycle cannot involve any slicing triangle, so only piercing triangles matter. As the number of piercing triangles is small, this ensures that the recurrence that controls the efficiency of our procedure solves to a bound close to $n^{3/2}$, as desired.

This comes at the expense of possibly somewhat raising the number of cuts along each triangle (but, fortunately, with no real increase in the asymptotic bound).

Alternative Algorithmic Approaches  We note that the previous study [7] proposes two other algorithmic approaches for computing the cuts in the case of lines: one using the algorithms of Har-Peled and Sharir [19] or of Solan [23], and the other using the (slower, but still polynomial, and sharper) approximation algorithm of Aronov et al. [3]. Unfortunately, neither of these alternative techniques seems, so far, applicable to the case of triangles.

In the Solan/Har-Peled–Sharir approach for cycle-cutting among lines or line segments, there are two crucial observations that make the algorithms correct and somewhat efficient (or at least subquadratic); we would need their analogues for triangles. One would be an efficient procedure for testing if a set of triangle fragments has a proper (cycle-free) depth order, analogous to the $O(n^{3/2+\varepsilon})$-time algorithm of [12]...
for segments; we are not aware of any such procedure with subquadratic running time, though there is a trivial quadratic-time test. Secondly, one needs to prove a connection between the size of the optimal partitioning and that of the partitioning obtained by an algorithm based on constructing a cutting of the projection of the triangles, analogous to the corresponding fact for segments, proven in [23]. That this connection exists is not apparent to us, at this point. Perhaps one can obtain such a connection by combining the above techniques with the methods of de Berg [10].

Finally, the approach of Aronov et al. [3] relies on a close relationship between the size of the minimum set of cuts for breaking cycles among line segments and the size of the minimum set of feedback vertices in a suitable directed graph. This relationship does not appear to have an obvious analogue in the case of triangles.

2 Preliminaries

2.1 The Polynomial Partitioning

For a non-zero polynomial \( f \in \mathbb{R}[x, y, z] \) of degree at most \( D \), we let

\[
Z(f) := \{(x, y, z) \mid f(x, y, z) = 0\}
\]

denote its zero set. We refer to the connected components of \( \mathbb{R}^3 \setminus Z(f) \) as cells. A main tool of our construction is the following result, which is a special case of a much more general Theorem 0·3 in [17], see also Guth and Katz [18]

Proposition 2.1 (Polynomial Partitioning for Lines; Guth [17]) Given a set of \( N \) lines in \( \mathbb{R}^3 \) and an integer \( 1 \leq D \leq \sqrt{N} \), there exists a non-zero polynomial \( f \in \mathbb{R}[x, y, z] \) of degree at most \( D \), such that each of the \( O(D^3) \) cells of \( \mathbb{R}^3 \setminus Z(f) \) intersects at most \( \frac{cN}{D^2} \) of the given lines, for some absolute constant \( c \).

Very recently, Agarwal et al. [1, Thm. 4] developed an efficient algorithm for constructing such a polynomial partition:

Proposition 2.2 (Constructive Polynomial Partitioning for Lines) A partitioning with the above properties can be constructed in expected time \( nD^{O(1)} + e^{D^{O(1)}} \).

Remark (a) The computation can be done in expected time \( O(n) \) for constant \( D \). (b) The work in [1] uses the Real RAM algebraic model of computation, where arithmetic operations on algebraic numbers can be performed at unit cost and identifying the roots of a degree-\( d \) univariate polynomial takes time that depends only on \( d \).

2.2 The Vertical Decomposition in an Arrangement of Algebraic Surfaces

In this section, we give, for completeness, a brief description of the notion of the vertical decomposition of \( \mathbb{R}^3 \) with respect to a set \( F \) of \( k \) algebraic surfaces, each of degree at most \( D \). We follow the definitions in [22, Sect. 8.3]. An illustration
of the analogous notion of a vertical decomposition (also known as a trapezoidal decomposition [11]) in the plane is shown in Fig. 4.

The vertical decomposition \( \mathcal{V} = \mathcal{V}(\mathcal{F}) \), for a collection \( \mathcal{F} \) as above, is constructed by first drawing a collection \( \Gamma \) of curves on the surfaces, where each curve is either the intersection of two surfaces of \( \mathcal{F} \), or is the locus of the singular points or of the points of \( z \)-vertical tangency on a single surface. We then take each curve \( \gamma \in \Gamma \) and erect from it a \( z \)-vertical “wall,” which is the union of all the maximal \( z \)-vertical segments that pass through \( \gamma \) and whose relative interiors do not meet any surface of \( \mathcal{F} \). That is, we extend each of the segments up and down until it meets a different point on some surface, or else all the way to \( z = \pm \infty \). The resulting decomposition has the property that each of its 3-cells \( \nu \) is \( xy \)-monotone (i.e., it intersects any \( z \)-vertical line, if at all, in a connected segment), as trivially follows from the construction in [22, Sect. 8.3], but the \( xy \)-projection \( \nu' \) of \( \nu \) need not be \( x \)-monotone, nor even simply connected. In the next step, we form the vertical decomposition of the projection \( \nu' \) of each first-stage 3-cell \( \nu \), by drawing a \( y \)-vertical segment from each singular point and from each point of \( y \)-vertical tangency on the boundary \( \partial \nu' \) of \( \nu' \), and by extending each such segment up and down, in the \( y \)-direction, within \( \nu' \), until it meets another point of \( \partial \nu' \), or else all the way to \( y = \pm \infty \). This yields a decomposition of \( \nu' \) into vertical pseudo-trapezoids. We lift each such pseudo-trapezoid \( \tau' \) in the \( z \)-direction (formally, by taking the Cartesian product with \( \mathbb{R} \)) and intersect the resulting unbounded prism \( \tau \cap \nu \). The resulting collection of pseudo-prisms and their boundaries constitutes \( \mathcal{V}(\mathcal{F}) \). For more details, see [22, Sect. 8.3].

The decomposition \( \mathcal{V}(\mathcal{F}) \) has the property that each of its open 3-cells is an open topological cube, given by up to six inequalities of the form \( a < x < b, \ f_1(x) < y < g_1(x), \) and \( f_2(x, y) < z < g_2(x, y), \) for continuous algebraic functions \( f_1, g_1, f_2, g_2, \) such that \( f_1(x) < g_1(x) \) for all \( x \in (a, b) \), and \( f_2(x, y) < g_2(x, y) \) for all \( x \in (a, b), y \in (f_1(x), g_1(x)) \). Each of \( a, f_1, \) or \( f_2 \) may also be \( -\infty \), and each of \( b, g_1, \) or \( g_2 \) may be \( +\infty \), in which case the corresponding inequality is dropped.

What about the degrees of the polynomials defining the resulting objects? If we start with surfaces that are zero sets of polynomials of degree at most \( D \), both the intersection curves and the curves covering critical points are obtained as the common
zeros of two polynomials of degree at most $D$ each and therefore, by Fulton [16], the intersection curve has degree $O(D^2)$. By Harris [20], projection does not increase the degree. Therefore all objects of $\mathcal{V}(\mathcal{F})$ can be described by polynomials of degree at most $O(D^2)$.

The above description focuses on 3-cells of $\mathcal{V} = \mathcal{V}(\mathcal{F})$ and does not specify the exact definition for its 2-, 1-, and 0-faces. It turns out that, for our purposes, such a description is largely unimportant. Instead of referring to the individual 2-, 1-, and 0-faces, we refer to the 2-

skeleton $\mathcal{V}^{(2)}$ of the vertical decomposition, defined as the complement of the union of its open 3-cells. By construction, every surface in $\mathcal{F}$ is contained in $\mathcal{V}^{(2)}$. This completes this brief description of the vertical decomposition. Again, see [22].

### 2.3 A Technical Topological Lemma

**Setup** We will need a technical lemma in our construction below. It involves an open three-dimensional cell of a vertical decomposition, as described above. The topological material needed for the lemma is fairly standard; it is covered, i.e., in Spanier [24]. We start with some definitions.

Fix real numbers $a, b \in \mathbb{R}$ with $a < b$. Let $f_1, g_1 : (a, b) \to \mathbb{R}$ be continuous functions satisfying $f_1(x) < g_1(x)$ for all $x \in (a, b)$. Define the open set $P \subset \mathbb{R}^2$ as

$$P := \{(x, y) \mid x \in (a, b), f_1(x) < y < g_1(x)\}.$$ 

Let $f_2, g_2 : P \to \mathbb{R}$ be continuous functions satisfying $f_2(x, y) < g_2(x, y)$ for all $(x, y) \in P$. Define the open set $\nu$ in $\mathbb{R}^3$ by

$$\nu := \{(x, y, z) \mid (x, y) \in P, f_2(x, y) < z < g_2(x, y)\}.$$ 

Each bounded open 3-cell $\nu$ of the vertical decomposition $\mathcal{V}$ of any collection of algebraic surfaces, as discussed in Sect. 2.2, is of the above form. Note that $\nu$ is homeomorphic to the open cube $(0, 1)^3$ under the continuous bijection

$$\Phi(x, y, z) = \left(\frac{x - a}{b - a}, \frac{y - f_1(x)}{g_1(x) - f_1(x)}, \frac{z - f_2(x, y)}{g_2(x, y) - f_2(x, y)}\right) \in (0, 1)^3,$$

for $(x, y, z) \in \nu$. In particular, since $\nu$ is homeomorphic to an open cube, it is path-connected and has vanishing integral homology. Finally let $L : \mathbb{R}^2 \to \mathbb{R}$ be any continuous function, and let the non-vertical surface $\Pi = \Pi(L)$ be the graph of $L$, namely

$$\Pi := \{(x, y, L(x, y)) \mid (x, y) \in \mathbb{R}^2\}.$$ 

Note that $\Pi$, with its relative topology as a subset of $\mathbb{R}^3$, is homeomorphic to $\mathbb{R}^2$ under the bijective projection $\rho : \mathbb{R}^3 \to \mathbb{R}^2$ given by $(x, y, z) \mapsto (x, y)$.

---

3 Unbounded cells are handled similarly; see the end of the proof of Lemma 2.4.
A useful point-set topology fact which follows from the local path connectedness of $\mathbb{R}^2$ is:

**Fact 2.3** Any open set $U \subset \mathbb{R}^2$ is a disjoint union of path-connected components, each of which is an open subset of $U$ and open in $\mathbb{R}^2$.

Let $\nu$ and $\Pi$ be as above, and assume that the intersection $\nu \cap \Pi$ is non-empty. Since $\nu$ is open in $\mathbb{R}^3$, $\nu \cap \Pi$ is open in the subspace topology of $\Pi$. Since $\Pi$ is homeomorphic to the 2-plane $\mathbb{R}^2$ by $\rho$, this intersection $\nu \cap \Pi$ (by Fact 2.3) is a disjoint union of path-connected open sets of $\Pi$. Let $V$ be one such component. A crucial step in our analysis, presented in Sect. 4, is the following seemingly elementary lemma:

**Lemma 2.4** The difference set $\nu \setminus V$ is an open set in $\mathbb{R}^3$ with exactly two open path-connected components. In particular, any continuous path in $\nu$ connecting points in different components of $\nu \setminus V$ must pass through $V$.

In the example of a donut in $\mathbb{R}^3$, one may make a cut of the above kind which does not disconnect it (as in Fig. 5), so one must invoke some property of (a space homeomorphic to) the open 3-cube (as is our set $\nu$) to avoid such a situation. Indeed, the crucial issue is that the open 3-cube carries no non-zero cycles, while the donut does. That is, the first singular homology with integer coefficients of the open cube is zero: $H_1((0, 1)^3; \mathbb{Z}) = H_1(\nu; \mathbb{Z}) = 0$; see below.

Informally, if $\nu \cap \Pi$ is connected, so that $V = \nu \cap \Pi$, the lemma is easy: Removal of $\Pi$ splits $\nu$ into one part “below $\Pi$” and one part “above $\Pi$;” the two parts cannot be connected by a path in $\nu$ without crossing $\Pi$, which follows from the Intermediate Value Theorem (proving that the number of resulting pieces is exactly two takes another argument). The argument is less immediate if $V$ is only one of several connected components of $\nu \cap \Pi$; note that $V$ need not be simply connected.

We will use singular homology groups with integer coefficients to prove our result. An open set $U \subset \mathbb{R}^d$ is path-connected if and only if the zeroth reduced singular homology vanishes, i.e., $\tilde{H}_0(U; \mathbb{Z}) = 0$. Also such an open set $U$ is the disjoint union of two open path-connected components if and only if the zeroth reduced singular homology equals $\mathbb{Z}$, i.e., $\tilde{H}_0(U; \mathbb{Z}) = \mathbb{Z}$. Hence, a proof of Lemma 2.4 is most naturally expressed in these singular homology terms.
**Proof of Lemma 2.4**  As we just observed, to prove the lemma, it is sufficient to show that $v \setminus V$ is open and that $\hat{H}_0(v \setminus V; \mathbb{Z}) = \mathbb{Z}$.

To see that $v \setminus V$ is open, consider the open set $v \cap \Pi$ in the topological space $\Pi$. By Fact 2.3, it is the disjoint union of open path-connected sets, one of which is the chosen $V$. Denote the union of the other components as (the open set) $W \subset \Pi$; $W$ may be empty. Since $\Pi$ has the subspace topology in $\mathbb{R}^3$, there is an open set $W^# \subset \mathbb{R}^3$ with $W = W^# \cap \Pi$. Hence,

$$v \setminus V = (v \setminus \Pi) \cup (W^# \cap v)$$

is a union of the two open sets $v \setminus \Pi$ and $W^# \cap v$ in $\mathbb{R}^3$ and hence open. (Note that $\Pi$ as a subset of $\mathbb{R}^3$ is closed, so $v \setminus \Pi$ is open in $\mathbb{R}^3$.)

Define $\hat{V} \subset \mathbb{R}^2$ as the image $\rho(V)$ of $V$ under the previously defined projection $\rho$. As $\rho$ restricted to $\Pi$ is a homeomorphism, $\hat{V}$ is an open set in $\mathbb{R}^2$ homeomorphic to $V$ and hence path-connected. Let $T$ denote the intersection

$$T := \rho^{-1}(\hat{V}) \cap v = \{(x, y, z) \mid f_2(x, y) < z < g_2(x, y) \text{ with } (x, y) \in \hat{V}\}.$$  

Then $T$ is the disjoint union of three subsets $V$, $X$, and $Y$ with the open sets $X$, $Y$ in $\mathbb{R}^3$ defined by

$$X := \{(x, y, z) \mid (x, y) \in \hat{V}, \; f_2(x, y) < z < L(x, y)\}, \quad \text{and}$$

$$Y := \{(x, y, z) \mid (x, y) \in \hat{V}, \; L(x, y) < z < g_2(x, y)\}.$$  

The open sets $X$, $Y$ are homeomorphic to $\hat{V} \times (0, 1)$ via the bijections

$$(x, y, z) \mapsto \left(x, y, \frac{z - f_2(x, y)}{L(x, y) - f_2(x, y)}\right) \quad \text{and}$$

$$(x, y, z) \mapsto \left(x, y, \frac{z - L(x, y)}{g_2(x, y) - L(x, y)}\right),$$

respectively. Since $\hat{V}$ is path-connected, so are $X$ and $Y$.

In these terms $v = (v \setminus V) \cup T$ with $(v \setminus V) \cap T = X \sqcup Y$, with $\sqcup$ representing a disjoint union of open sets. The associated zeroth singular reduced homologies, by path-connectivity, are $\hat{H}_0(v; \mathbb{Z}) = 0$, $\hat{H}_0(X; \mathbb{Z}) = 0$, $\hat{H}_0(Y; \mathbb{Z}) = 0$, and $\hat{H}_0(X \sqcup Y; \mathbb{Z}) = \mathbb{Z}$ where the generator of the latter is represented by the 0-chain $c_0 = 1 \cdot p - 1 \cdot q$ for any pair of chosen points $p \in X$, $q \in Y$ (see [24] for details). Moreover, since $\rho$ is homeomorphic to an open 3-cube, $H_1(v; \mathbb{Z}) = 0$ as well.

To finish the proof that $\hat{H}_0(v \setminus V; \mathbb{Z}) = \mathbb{Z}$, recall from Spanier [24, Thm. 4.6.3, p. 188], that if $X_1 \cup X_2$ is a topological space which is the union of two open sets $X_1$, $X_2$, then there exists the following long exact sequence (the *Mayer–Vietoris sequence*):

---

4 $\Pi = \rho^{-1}(\mathbb{R}^2)$, $\mathbb{R}^2$ is a closed subset of itself, and $\rho$ is continuous, hence $\Pi$ is closed.
\[ H_1(X_1 \cup X_2; \mathbb{Z}) \to \tilde{H}_0(X_1 \cap X_2; \mathbb{Z}) \]
\[ \to \tilde{H}_0(X_1; \mathbb{Z}) \oplus \tilde{H}_0(X_2; \mathbb{Z}) \to \tilde{H}_0(X_1 \cup X_2; \mathbb{Z}). \]

Applied to the case of \( X_1 = T \) and \( X_2 = v \backslash V \), so \( X_1 \cup X_2 = v \) and \( X_1 \cap X_2 = X \cup Y \), this long exact sequence reads
\[ H_1(v; \mathbb{Z}) \to \tilde{H}_0(X \cup Y; \mathbb{Z}) \to \tilde{H}_0(T; \mathbb{Z}) \oplus \tilde{H}_0(v \backslash V; \mathbb{Z}) \to \tilde{H}_0(v; \mathbb{Z}), \]
\[ \{0\} \to \mathbb{Z} \to \{0\} \oplus \tilde{H}_0(v \backslash V; \mathbb{Z}) \to \{0\}. \]

Consequently, since \( v \backslash V \) is an open set in \( \mathbb{R}^3 \), it follows from the isomorphism \( \mathbb{Z} \cong \tilde{H}_0(v \backslash V; \mathbb{Z}) \), implied by the above sequence, that \( v \backslash V \) is a disjoint union of two non-empty open path-connected sets \( X^\# \) and \( Y^\# \). Since the image of the 0-chain \( c_0 = 1 \cdot p - 1 \cdot q \) represents the generator of \( \tilde{H}_0(v \backslash V; \mathbb{Z}) \), one of these path-connected open sets, say \( X^\# \), contains \( p \) and other set \( Y^\# \) contains \( q \). Hence, necessarily \( X \subset X^\# \) and \( Y \subset Y^\# \). Therefore
\[ v \backslash V = X^\# \sqcup Y^\#, \]
with \( X^\# \), \( Y^\# \) open disjoint path-connected sets in \( \mathbb{R}^3 \). Since \( v \) is path-connected, any two points \( p' \in X^\# \) and \( q' \in Y^\# \) are joined by a continuous path within \( v \). This path necessarily must pass through \( V \) as the 1-chain \( 1 \cdot p' - 1 \cdot q' \) is homologous to \( 1 \cdot p - 1 \cdot q \) and the latter represents the generator of \( \mathbb{Z} = \tilde{H}(v \backslash V; \mathbb{Z}) \), since \( X^\# \), \( Y^\# \) are path-connected.

This completes the proof of the lemma when \( v \) is bounded. The same proof applies verbatim, with only minor notational inconveniences, to unbounded \( v \): Roughly, \( a, b, f_1, f_2, g_1, \) and/or \( g_2 \) must be replaced by \( \pm \infty \), as appropriate, and the “open cube”—by a suitable unbounded open box homeomorphic to \( \mathbb{R}^3 \).

\[ \square \]

### 3 Depth Cycles in a Set of Triangles in \( \mathbb{R}^3 \)

We now begin our description of the actual process of cutting triangles so as to eliminate cycles in the depth relation among the resulting pieces. We start with some initial notation and then state our problem more formally.

#### The Setup and Some Notation
Let \( O \) be a collection of pairwise disjoint objects in three dimensions, where each object is a relatively open path-connected subset of a non-vertical plane. For most of the paper, the objects will be the triangles or the triangle pieces produced by our construction. We choose to treat the objects as relatively open in order to make the vertical relation unambiguous in situations where two objects touch, such as two triangles sharing an edge or two adjacent pieces of the same triangle. This allows us to consistently treat portions into which a triangle is cut as properly disjoint.

Clearly, each object in \( O \) is xy-monotone, that is, its intersection with any vertical line is a single point or empty. We define the depth relation \((O, \prec)\) on the objects of \( O \), extending the way we defined it for triangles: \( o_1 \in O \) lies (or passes) below
\( o_2 \in \mathcal{O} \) (in which case we also say that \( o_2 \) passes above \( o_1 \)), and write \( o_1 < o_2 \) or \( o_2 > o_1 \), if there exists a vertical line \( \ell \) that meets both \( o_1 \) and \( o_2 \), and the \( z \)-coordinate of its intersection with \( o_1 \) is smaller than that of its intersection with \( o_2 \). For arbitrary disjoint shapes, it is possible for \( o_1 < o_2 \) and \( o_1 > o_2 \) to hold simultaneously, but we will be focusing on objects that come from cutting disjoint convex shapes such as triangles into smaller pieces, so this situation will not arise; this will not affect any of our arguments below, however.

A set \( X \) is semialgebraic if it can be expressed as the locus of points satisfying a Boolean formula with atoms of the form \( P = 0 \) or \( P < 0 \), with \( P \) a polynomial. The complexity of \( X \) is bounded by \( b \) if the formula involves at most \( b \) variables, the polynomials have degree at most \( b \), and the formula length is at most \( b \).

A cycle in \( (\mathcal{O}, <) \) is a circular sequence of some \( k \) objects from \( \mathcal{O} \) that satisfy \( o_1 < o_2 < \cdots < o_k < o_1 \). We refer to \( k \) as the length of the cycle; a cycle of length \( k \) is a k-cycle. Note that self-loops and 2-cycles are not possible in \( \mathcal{O} \) under our assumptions (although they may very well exist for more general objects, already for algebraic arcs), so we must have \( k \geq 3 \).

Let \( \mathcal{O} \) be a collection of objects with a cycle \( C \) in \( (\mathcal{O}, <) \) as above. We associate with \( C \) a continuum \( \Pi(C) = \Pi(\mathcal{O}, C) \) of closed paths (to which we refer as loops realizing \( C \)), where, informally, each loop \( \pi \) in \( \Pi(C) \) traces the cycle along the objects. Formally, each such \( \pi \) is defined in terms of \( k \) vertical lines \( \ell_1, \ldots, \ell_k \), such that, for each \( i \), the line \( \ell_i \) intersects both \( o_i \) and \( o_{i+1} \) (where addition of indices is mod \( k \)), at respective points \( v_i^+, v_{i+1}^- \), so that \( v_i^- \) lies below \( v_{i+1}^+ \). For each \( i \), we connect the two points \( v_i^+, v_i^- \in o_i \) by a Jordan arc \( \pi_i \subset o_i \). The loop \( \pi \) is then the cyclic concatenation

\[
\pi = \pi_1 \| v_1^- v_2^+ \| \pi_2 \| v_2^- v_3^+ \| \cdots \| v_{k-1}^- v_k^+ \| \pi_k \| v_k^- v_1^+, \n\]

which is an alternation between the arcs \( \pi_i \) along the objects, and the upward vertical jumps \( v_i^- v_{i+1}^+ \) between them. As already said, there is a continuum of possible loops realizing \( C \), representing different choices of the vertical lines (and thus points) at which we decide to jump from object to object, and of the paths along which the “landing” and “take-off” points are connected along each object. (This is in stark contrast to the case of lines, studied in [7], where a unique loop realizes each cycle, as long as the lines are in general position.)

**The Problem, Restated** We are now ready to formally state the problem. Let \( \mathcal{T} \) be a collection of \( n \) non-vertical pairwise disjoint relatively open triangles in \( \mathbb{R}^3 \). We assume, for simplicity of presentation, that no triangle lies in a vertical plane to avoid some technicalities which do not essentially affect the rest of our argument. As illustrated in Fig. 1, the depth relation \( (\mathcal{T}, <) \) may contain cycles. Our goal is to cut the triangles of \( \mathcal{T} \) into a small number of relatively open path-connected semialgebraic pieces so that, for the collection \( \mathcal{O} \) of the resulting pieces, the depth relation \( (\mathcal{O}, <) \) is acyclic—a depth order.

As already mentioned, a classical method for eliminating all cycles is provided by the binary space partition (BSP) technique of Paterson and Yao [21], which removes all cycles by cutting the triangles into \( O(n^2) \) triangular pieces. Indeed, the construction in [21] has a much stronger property: the resulting collection of triangular pieces has no
cycles in the depth relation corresponding to any viewing direction, or, more generally, to the perspective view from any point.

We will cut the triangles by drawing curves on each of them; this will be performed in a hierarchical manner, by a recursive algorithm. For each triangle $\Delta$, the curves drawn on $\Delta$ form a planar arrangement within $\Delta$, and the overall collection of faces of these arrangements, over all $\Delta \in T$, will have an acyclic depth relation. In general, though, these faces need not have a constant number of edges nor be simply connected, so a final step breaks them into pseudo-trapezoidal subfaces that have at most four edges each, without affecting the asymptotic bound on the number of pieces.

To eliminate all cycles, it suffices to cut all the loops realizing them, in a manner made precise in the following definitions and easy lemma.

For each $\Delta \in T$, let $\Gamma_\Delta$ be a finite collection of algebraic curves drawn on $\Delta$. Let $O_\Delta$ denote the collection of relatively open two-dimensional faces of the arrangement $A(\Gamma_\Delta)$ contained in $\Delta$. Put $\Gamma := \bigcup_\Delta \Gamma_\Delta$ and $O := \bigcup_\Delta O_\Delta$. In words, $O$ is the collection of triangle pieces resulting from cutting the triangles in $T$ by the curves of $\Gamma$.

We say that a loop $\pi$ realizing a depth cycle is cut by a curve of $\Gamma$ if one of its closed arcs $\pi_i$ is cut, i.e., if it meets at least one of the curves of $\Gamma$. Equivalently, $\pi_i \subset \Delta$ is cut if it is not fully contained in one of the faces of $O_\Delta$; $\pi$ is cut if one of the paths $\pi_i$ is cut.

**Lemma 3.1** In the above terminology, to verify that the depth relation among the pieces in $O$ is acyclic, it is sufficient to ensure that, for each cycle $C$ in $(T, \prec)$, and for each loop $\pi \in \Pi(C, T)$, $\pi$ is cut by a curve in $\Gamma$.

**Remark** Notice that we require that all loops in $\Pi(C, T)$ be cut. Indeed, for a specific loop $\pi$ realizing $C$, a subpath $\pi_i$ may be cut in such a manner that it first leaves and then reenters the same face of $A(\Gamma_\Delta)$. This by itself does not eliminate $C$, as $\pi_i$ can be replaced by a rerouted subpath $\pi_i'$ that stays within the same face. However, replacing $\pi_i$ by $\pi_i'$ in $\pi$ produces a different loop in $\Pi(C, T)$, which we also require to be cut.

**Proof** We proceed by contradiction: Assume that all loops in $\Pi(C, T)$, for every cycle $C$ in $(T, \prec)$, have been cut, but nonetheless there remains a cycle $C'$: $o_1 \prec o_2 \prec o_3 \prec \cdots \prec o_k \prec o_1$ in $(O, \prec)$. In this case the set $\Pi(C', O)$ of loops realizing $C'$ is non-empty, and we pick a loop $\pi \in \Pi(C', O)$ realizing $C'$, where each subpath $\pi_i$ is contained in the corresponding piece $o_i \in O$, and each vertical jump $v_i^- v_i^+$ moves from $o_i$ to $o_{i+1}$. Each $o_i$ is contained in some, not necessarily distinct triangle $\Delta_i \in T$, so $\pi_i$ is fully contained in $\Delta_i$, the jump $v_i^- v_i^+$ can be viewed as a vertical jump from $\Delta_i$ to $\Delta_{i+1}$, and therefore $\pi$ realizes the cycle $\Delta_1 \prec \Delta_2 \prec \cdots \prec \Delta_k \prec \Delta_1$ in $(T, \prec)$, and has not been cut, contradicting our assumption. \hfill $\square$

## 4 Eliminating Depth Cycles in a Set of Triangles

### 4.1 The Procedure for Cutting the Triangles

In this section we present and analyze the procedure for eliminating depth cycles in a set of triangles in three dimensions. The procedure is recursive. At each step $\xi$ of the recursion we have an open connected region $\sigma_\xi$ and the subset $T_\xi$ of the triangles of $T$ that pierce $\sigma_\xi$. We put $n_\xi := |T_\xi|$. We apply the following steps.

\[\text{Springer}\]
We construct a partitioning polynomial \( f_\xi \), as in Proposition 2.1, for the \( 3n_\xi \) lines supporting the edges of the triangles of \( T_\xi \); without loss of generality, we assume that \( f_\xi \) is square-free. We put \( F_\xi = F_\zeta f_\xi \), where \( F_\zeta \) is the product polynomial \( \prod_{i=1}^s f_{\xi_i} \) and \( \xi_1, \ldots, \xi_s \) are the proper recursive ancestors of \( \xi \), so that \( \xi_1 \) is the root and \( \xi_s = \xi \) is the parent of \( \xi \). We have \( F_{\text{root}} = 1 \).

(b) We generate three types of curves (traces, critical shadows, and wall shadows, see the detailed description below) on the triangles of \( T_\xi \), with up to \( O(D^4 \log^2 n) \) curves, each of degree up to \( O(D^2) \), per triangle. Each curve is first clipped to within the closure of \( \sigma_\xi \).

(c) For each cell \( \sigma \) of \( \mathbb{R}^3 \setminus Z(F_\xi) \) that is contained in \( \sigma_\xi \), we generate a recursive subproblem \( \eta \), put \( \sigma_\eta := \sigma \), and let \( T_\eta \) be the subset of triangles of \( T_\xi \) that pierce \( \sigma \).

(d) The bottom of the recursion is at nodes \( \xi \) for which \( |T_\xi| \leq D^2/c \), where \( c \) is the modified constant in Proposition 2.1. For such cells we apply the Paterson–Yao binary space partitioning [21], which cuts the triangles into \( O(|T_\xi|^3) = O(D^6) \) triangular pieces, whose depth relation does not contain cycles, and retains only the portions of those pieces within \( \sigma_\xi \). More precisely, we add the straight segments bounding the triangular pieces to the collections of curves on the triangles (clipping them to within \( \sigma_\xi \)).

We now spell out the details of step (b), as the remaining steps have been fully specified. We draw curves of the following three types.

**Traces** For each triangle \( \Delta \in T_\xi \) not fully contained in \( Z(f_\xi) \), we draw \( Z(f_\xi) \cap \Delta \) on \( \Delta \). We call this the trace of \( f_\xi \) on \( \Delta \). It is a curve of degree at most \( D \). If \( \Delta \subset Z(f_\xi) \), that is, if the plane \( h_\Delta \) supporting \( \Delta \) is a component of \( Z(f_\xi) \), we do not draw any such curve on \( \Delta \).

**Critical Shadows** We next consider the set of points \( p \in Z(f_\xi) \) that are either singular or at which \( Z(f_\xi) \) has a \( z \)-vertical tangent line. For simplicity, we approximate it with the possibly larger superset \( S_\xi := Z(f_\xi, \partial f_\xi / \partial z) \), the common zero set of \( f_\xi \) and \( \partial f_\xi / \partial z \); recall that \( f_\xi \) is square-free.

Consider the union of all vertical lines contained in \( Z(f_\xi) \); this set can be written as \( G_\xi \times \mathbb{R} \), for some (maximal, at most one-dimensional) \( G_\xi \subset \mathbb{R}^2 \). If \( G_\xi \) is one-dimensional, \( f_\xi \) must have a factor that does not depend on \( z \). Intuitively, \( S_\xi \) consists of two parts: One part is the “vertical curtain” \( G_\xi \times \mathbb{R} \subset S_\xi \). The remainder of \( S_\xi \), roughly speaking, consists of singular points and/or points of “proper” vertical tangency of \( Z(f_\xi) \) not lying on vertical lines contained in \( Z(f_\xi) \). This second part of \( S_\xi \) is at most one-dimensional.

Let \( H_\xi \) denote the vertical curtain induced by \( S_\xi \), namely, the union of all vertical lines that pass through points of \( S_\xi \). Since \( S_\xi \) is an algebraic variety of degree \( O(D^2) \) (and only two-dimensional portions of \( S_\xi \), if any, are vertical curtains), \( H_\xi \) is a two-dimensional variety of the same degree (see, e.g., [16]). The equation for \( H_\xi \) is obtained by eliminating \( z \) from the system \( f_\xi = \partial f_\xi / \partial z = 0 \). Note that \( G_\xi \times \mathbb{R} \subset H_\xi \). We then draw, on each triangle \( \Delta \in T_\xi \), including triangles contained in \( Z(f_\xi) \), the critical shadow curve \( H_\xi \cap \Delta \). It is a curve of degree \( O(D^2) \).
Wall Shadows We eventually want to proceed recursively, within each cell $\sigma$ of the partition, but before doing so, we discuss in more detail the notions of piercing and slicing triangles, as already defined, to set the stage for the last type of curves that we draw.

Consider the interaction of the triangles $\Delta \not\subset Z(f_\xi)$ with the cells of the partition. By construction, each cell $\sigma$ meets the edges of at most $O(n_\xi/D^2)$ triangles. However, $\Delta \in T_\xi$ may meet $\Theta(D^2)$ cells in the worst case, a consequence of Warren’s theorem [27]. Therefore, each cell $\sigma$ meets $O(n_\xi/D^2)$ triangle edges and, on average, $O(n_\xi D^2/D^3) = O(n_\xi/D)$ triangle interiors; in the worst case, the latter average bound is tight. Roughly speaking, the latter quantity is too large and yields an unfavorable recurrence, so we have to be more careful: Our plan now is to recurse, for each cell $\sigma = \sigma_\eta$, where $\eta$ is a child of $\xi$, only on the set $T_\eta \subset T$ of triangles that pierce it, and disregard, for the purposes of recursion, the triangles of $T_\xi$ that slice it. However, this is safe only if the slicing triangles do not participate in any uncut loop realizing a depth cycle and is fully contained within $\sigma$. We draw additional curves on the triangles, as described next, to ensure this.

We construct the vertical decomposition $\mathcal{V} := V(Z(F_\xi))$ of 3-space produced by $Z(F_\xi)$; see Sect. 2.2 for the variant of vertical decomposition we need here. Recall that $F_\xi$ is a product of $k := O(\log_D n)$ polynomials, each of degree at most $D$, so we actually construct the vertical decomposition of the arrangement $\mathcal{A}(\mathcal{F})$ of the collection $\mathcal{F}$ of the zero sets of these $k$ polynomials. We retain only the portion of $\mathcal{V}$ within the closure of $\sigma_\xi$ and by a slight abuse of notation continue to refer to the resulting object as $\mathcal{V}$; by the properties of vertical decomposition (and since we construct it for $F_\xi$ rather than for $f_\xi$) it is easy to verify that each three-dimensional cell of $\mathcal{V}$ is either fully contained in $\sigma_\xi$ or disjoint from it. Recall that the 2-skeleton $\mathcal{V}^{(2)}$ of $\mathcal{V}$ is the complement (within the closure of $\sigma_\xi$, here) of the union of its open three-dimensional cells.

We are now ready to draw our third kind of curves, called wall shadows, on the triangles of $T_\xi$. Fix $\Delta \in T_\xi$. For every open three-dimensional cell $\nu$ of $\mathcal{V}$ meeting $\Delta$, we draw the one-dimensional boundary of $\nu \cap \Delta$ on $\Delta$. It is also possible for the closure $\bar{\nu}$ of an open three-dimensional cell $\nu$ of $\mathcal{V}$ to meet $\Delta$ without the cell itself meeting $\Delta$. This happens, for example, when $\Delta \subset Z(f_\xi)$ is part of the floor or ceiling of $\nu$. In this case, we draw the boundary of the floor/ceiling of $\nu$ on $\Delta$. Other scenarios where $\Delta$ meets $\bar{\nu}$ but not $\nu$ arise when $\Delta \cap \bar{\nu}$ is 0- or 1-dimensional. In these cases $\nu$ does not cause any curves to be drawn on $\Delta$, but neighboring cells will generate such curves.

Implications As described in Sect. 2.2, $\mathcal{V}$ is based on a collection of curves drawn on the surfaces of $\mathcal{F}$, which are the intersection curves of pairs of the surfaces, and the loci of singular points and points with $z$-vertical tangencies on the individual surfaces. There is a total of $O(k^2) = O(\log_D^2 n)$ such curves, each of degree $O(D^2)$. These curves define the vertical walls of the first-stage prisms of $\mathcal{V}$, and these walls are thus vertical surfaces of degree $O(D^2)$ too. Consequently, the wall shadows that these surfaces form on a triangle $\Delta$ consist of $O(\log_D^2 n)$ curves, each of degree $O(D^2)$.

The second stage of the decomposition creates additional planar vertical walls, each contained in a vertical plane orthogonal to the $x$-axis. The resulting wall shadows that
these walls produce on our triangles are line segments. It is easy to obtain, from the
construction of \( V \), a somewhat crude upper bound of \( O(k^4D^4) = O(D^4 \log^4_D n) \) on
the number of these vertical walls, implying a bound of \( O(D^4 \log^4_D n) \) on the number
of these line segments within \( \Delta \). To recap, we have:

**Lemma 4.1** We draw, on each triangle of \( T_\xi \), at most \( O(\log^2_D n) \) curves of degree
\( O(D^2) \) and \( O(D^4 \log^4_D n) \) additional straight segments.

Let \( \nu \) be an open three-dimensional cell of \( V \) and let \( \Delta \in T \) be a triangle that slices \( \nu \)
(that is, as for the undecomposed cells of \( \mathbb{R}^3 \setminus Z(f_\xi) \), \( \Delta \) meets \( \nu \), but the boundary
of \( \Delta \) avoids \( \nu \)); note that \( \Delta \) may or may not belong to \( T_\xi \). Consider a loop \( \pi \) realizing
some depth cycle among portions of the original triangles (only loops fully contained
in \( \sigma_\xi \) need to be considered). Recall that such a loop consists of vertical upward jumps
alternating with connected arcs lying on triangles of \( T \). We say that \( \pi \) visits \( \Delta \) inside
\( \nu \) if its intersection with \( \Delta \cap \nu \) is non-empty. Notice that \( \pi \) necessarily arrives at \( \Delta \)
from a point lying below the plane \( h_\Delta \) supporting \( \Delta \) and leaves \( \Delta \) to a point above
that plane. (This visit might consist of a single point where some unrelated vertical
jump just crosses \( \Delta \).)

The following lemma is crucial for the correctness of our cycle-cutting procedure.
Its proof uses Lemma 2.4 and some terminology and notation from Sect. 2.3.

**Lemma 4.2** A loop \( \pi \) fully contained in a open cell \( \nu \) of \( V(Z(F_\xi)) \) within \( \sigma_\xi \)
cannot visit a triangle \( \Delta \) that slices \( \nu \).

**Proof** Let \( \Delta \in T \) be one of the triangles that slice \( \nu \), and let \( h_\Delta \) be its supporting plane.
For a contradiction, suppose that there exists a loop \( \pi \) realizing some depth cycle, so
that \( \pi \) is fully contained in \( \nu \) and visits \( \Delta \). (Note that, by construction, \( \pi \) meets \( \Delta \)
a finite number of times, in the sense that \( \pi^{-1}(\Delta) \) consists of a finite number of
connected components.) Let \( p \) be a point on \( \pi \) in \( \Delta \cap \nu \). Define \( L \) to be the linear
function whose graph \( \Pi \) (in the terminology of the proof of Lemma 2.4) is \( h_\Delta \). Let
\( V \) be the path-connected component of \( h_\Delta \cap \nu \) containing \( p \); since \( \Delta \) slices \( \nu \), this
component coincides with the path-connected component of \( \Delta \cap \nu \) containing \( p \).

Lemma 2.4 shows that \( \nu \setminus V \) consists of two path-connected components, the
“lower” \( X^\# \) and the “upper” \( Y^\# \), and any path in \( \nu \) passing from one component to
the other has to meet \( V \) between them. In particular, since \( \pi \) is a closed loop, it must
travel from \( X^\# \) to \( Y^\# \) (necessarily via \( V \)) as many times as it travels from \( Y^\# \) back
to \( X^\# \) (again, necessarily via \( V \)). However, by construction, \( \pi \) can only pass from
below \( V \subset \Delta \) to above it (which may be witnessed by a subpath \( \pi_i \) contained in
\( \Delta_i = \Delta \) and the corresponding nearby portions of \( \pi \) just before and after \( \pi_i \), or when
one of the upward jumps intersects \( V \)), and not in the opposite direction. This yields a
contradiction unless \( \pi \) avoids \( V \) entirely, thereby concluding the proof of the lemma.

\( \square \)

We now make the following observations which will be helpful in proving the
correctness of our procedure. In preparation for the following argument, as in [7],
define the level \( \lambda(q) \) of a point \( q \in \mathbb{R}^3 \) with respect to \( Z(f_\xi) \) to be the number of
intersection points of $Z(f_{\xi})$ with the relatively open downward-directed $z$-vertical ray $\rho_q$ emanating from $q$. Formally, if $q = (x_0, y_0, z_0)$, we consider the univariate polynomial $f_0(z) = f_{\xi}(x_0, y_0, z)$, and the level $\lambda(q)$ of $q$ is the number of real zeros of $f_0$ in $(-\infty, z_0)$, counted with multiplicity. At points $q$ where the entire vertical line through $q$ is contained in $Z(f_{\xi})$, so that $f_0 \equiv 0$, $\lambda(q)$ is undefined. We will explicitly deal with such points in the analysis below. The number of such lines is $O(D^2)$ (see below), unless $Z(f_{\xi})$ contains a “vertical curtain,” i.e., if $f_{\xi}$ has a factor that does not depend on $z$.

**Lemma 4.3** Consider a loop $\pi$ realizing a cycle $C$ among the triangles of $T_{\xi}$ and contained in $\sigma_{\xi}$, such that at least one of its open vertical jump segments meets $Z(f_{\xi})$. Then $\pi$ is cut by a curve created while processing $\xi$.

**Proof** Assume, without loss of generality, that the segment in question is $s := v_1^- v_2^+$. First, assume that $\lambda(\cdot)$ is defined at every point of $\pi$. In particular, $s$ is not fully contained in $Z(f_{\xi})$. We traverse $\pi$ in circular order, starting at $v_1^-$. At each point $q$ where $s$ intersects $Z(f_{\xi})$, the level increases as we traverse $s$ (upwards, as $\pi$ does) past this point.

Since (a) each of the vertical jumps $v_i^- v_{i+1}^+$ the level can never decrease, and (b) $\pi$ is a closed loop, the level has to come back to its original value at some point $q$ on one of the subpaths $\pi_i$ of $\pi$ (including possibly an endpoint of such a path). However, $\pi_i \subseteq \Delta_i \subseteq T_{\xi}$, and we cut $\Delta_i$ at all points where $Z(f_{\xi})$ meets $\Delta_i$ (by the corresponding trace), and along the critical shadow curves, which, collectively, are precisely the points where such an event may occur, completing the argument for the case where the level is defined everywhere on $\pi$.

If $s \subseteq Z(f_{\xi})$, the level is undefined on $s$. But then, $v_1^-$ and $v_2^+$ must be critical shadow points, and $\pi$ is cut at both of them. A similar argument applies if $\lambda$ becomes undefined at any other point along $\pi$. This completes the proof.  

**Lemma 4.4** Consider a depth cycle $C$ among the triangles of $T_{\xi}$ and a loop $\pi \in \Pi(C, T_{\xi})$ realizing it and contained in $\sigma_{\xi}$, such that one of its subpaths $\pi_i$ meets $\mathcal{V}^{(2)}$. Then $\pi$ is cut.

**Proof** By definition of $\pi$, we have $\pi_i \subseteq \Delta_i$, for some triangle $\Delta_i \in T_{\xi}$. We consider several ways in which $\pi_i$ may meet $\mathcal{V}^{(2)}$. We subdivide $\mathcal{V}^{(2)}$ into the relatively open floors and ceilings of cells of $\mathcal{V}$ and the relatively closed vertical walls of these cells; these two sets need not be disjoint, as a ceiling of a cell need not coincide with the floor of the cell above it, roughly speaking, and in fact may overlap the floors of several such cells.

If $\pi_i$ intersects a vertical wall, then it is cut at each of these points by a wall shadow drawn on $\Delta_i$. (Recall that we assumed that there are no vertical triangles in $T$, so $\Delta_i$ cannot overlap a vertical wall.) Note that $\pi_i$ may partially or completely overlap a wall shadow, but it is still cut in such cases because the drawn curves are removed in the formation of the final triangle pieces.

Now suppose $\pi_i$ avoids the relatively closed vertical walls of the cells in $\mathcal{V}$ and therefore only meets the overlap of a relatively open floor of such a cell with the relatively open ceiling of another cell (lying immediately below the first one). Recall
that the non-vertical surfaces of \( V^{(2)} \) (within the region \( \sigma_\xi \) of the current recursive instance) are contained in \( Z(\sigma_\xi) \).

If \( \Delta_i \) is not contained in \( Z(\sigma_\xi) \) then it is cut by the trace of \( Z(\sigma_\xi) \). In particular, \( \pi_i \subseteq \Delta_i \) must be cut at a point of \( \Delta_i \cap Z(\sigma_\xi) \). Otherwise, the plane \( h_{\Delta_i} \) supporting \( \Delta_i \) is fully contained in \( Z(\sigma_\xi) \). Then we made no cuts on \( \Delta_i \) by the trace of \( Z(\sigma_\xi) \) and \( \pi_i \) is fully contained in the relatively open floor of a cell \( \nu \) of \( V \). Follow \( \pi \) from \( \pi_i \) onward.

The vertical jump \( v_i^+ v_{i+1}^- \) following \( \pi_i \) must enter the open cell \( \nu \) and therefore leave \( Z(\sigma_\xi) \). In particular, \( \lambda(v_i^-) < \lambda(v_{i+1}^+ \) (the level cannot be undefined here, as is easily checked, as \( v_i^- v_{i+1}^+ \) passes through the open cell \( \nu \) of \( V \)). As in the preceding proof, since the level comes back to its value after a circular traversal of the loop \( \pi \), it must go back down or become undefined somewhere along the loop. The level can only go up along vertical jumps (unless it is undefined there). It thus follows, as before, that of those cuts along traces or critical shadows, which eliminate precisely those points on the triangles of \( T_\xi \) where the level changes or becomes undefined, will cut \( \pi \), as desired.

Finally, if the level is undefined along an open vertical jump, it is also undefined at its endpoints, which lie on triangles of \( T_\xi \), so \( \pi \) is also cut at such points. Having exhausted all cases, we have completed the proof. \( \square \)

### 4.2 All Cycles are Eliminated

Let \( \Gamma \) denote the set of all clipped curves that have been generated throughout the recursion. We write \( \Gamma \) as the disjoint union \( \bigsqcup_{\Delta} \Gamma_{\Delta} \), where \( \Gamma_{\Delta} \) is the set of curves drawn on \( \Delta \), for each \( \Delta \in T \).

Consider a node \( \xi \) of our recursive procedure, and let \( \zeta \) be its parent node (ignore \( \zeta \) when \( \xi \) is the root). Recall that each curve \( \gamma \in \Gamma \), constructed when partitioning the cell \( \sigma_\zeta \) corresponding to node \( \zeta \), is clipped to within \( \sigma_\zeta \), and may visit several subcells of that cell.

**Lemma 4.5** The procedure described above eliminates all the depth cycles in \( T_\xi \), in the sense that, for each cycle \( C \) in \((T, \prec)\) and for each loop \( \pi \in \Pi(C, T) \) realizing it, at least one of the “on-triangle” closed subpaths \( \pi_i \) of \( \pi \) meets a curve of \( \Gamma \).

**Proof** Let \( C : \Delta_1 \prec \Delta_2 \prec \cdots \prec \Delta_k \prec \Delta_1 \) be a cycle in \((T, \prec)\), and let \( \pi \) be a loop in \( \Pi(C, T) \) realizing \( C \). Let \( T_C = \{ \Delta_i \mid i = 1, \ldots, k \} \) be the set of triangles appearing in \( C \). Let \( \xi \) be the lowest (farthest from the root) node that satisfies the following two properties: (a) \( \pi \) is completely contained in the cell \( \sigma_\xi \) corresponding to \( \xi \) in the subdivision formed at its parent node \( \zeta \) (recall that \( \sigma_{\text{root}} = \mathbb{R}^3 \)), and (b) all the triangles in \( T_C \) are piercing for \( \zeta \) (this condition is vacuous when \( \xi \) is the root). Such a node \( \xi \) always exists and is unique. Indeed, the root satisfies (a) and, vacuously, (b), and the set of nodes satisfying both (a) and (b) is easily seen to form a contiguous path; \( \xi \) is the bottommost node of this path. We will ensure that \( \pi \) is cut when processing either \( \xi \) or its parent \( \zeta \), depending on which of the following two situations arises.

Each triangle \( \Delta_i \) meets \( \sigma_\xi \), so it is either a piercing triangle or a slicing triangle for \( \sigma_\xi \). We distinguish between two cases: (i) All triangles in \( T_C \) pierce \( \sigma_\xi \). (ii) At least one of these triangles slices \( \sigma_\xi \).
Consider first the situation in case (ii). Note that $\xi$ is not the root (no triangle slices $\sigma_{\text{root}} = \mathbb{R}^3$), and recall that $\zeta$ is its parent. We will show that, in case (ii), one of the subpaths $\pi_i$ of $\pi$ is cut by one of the curves drawn while processing $\xi$. We note that $\pi$ cannot be fully contained in any open three-dimensional cell $\nu$ of the vertical decomposition $V = V(Z(F_{\zeta}))$ within $\sigma_{\zeta}$. Indeed, if this were the case, as the vertical decomposition $\mathcal{V}$ is a refinement of the decomposition of $\sigma_{\zeta}$ induced by $Z(F_{\zeta})$, necessarily $\nu \subset \sigma_{\zeta}$ and therefore each triangle in $T_C$ that sliced $\sigma_{\zeta}$ would also slice $\nu$ (and there is at least one such triangle, by assumption), so $\pi$ would visit this triangle in $\nu$, but Lemma 4.2 asserts that this is impossible. It therefore must be the case that either $\pi$ meets an open three-dimensional cell $\nu$ of $\mathcal{V}$ but is not fully contained in $\nu$, or $\pi$ is fully contained in the boundary portion $\nu^{(2)}$ of $\mathcal{V}$. In the latter case, all of the subcurves $\pi_i$ of $\pi$ lie in $\nu^{(2)}$ and therefore $\pi$ is cut, by Lemma 4.4 applied to $\zeta$. Hence $\pi$ must meet $\nu$ without being contained in it, and without any of its subpaths $\pi_i$ meeting $\nu^{(2)}$, which is implied by another application of Lemma 4.4. This means that one of the open vertical jumps on $\pi$ must meet $\nu^{(2)}$ without meeting $Z(f_{\zeta})$. (If it did meet $Z(f_{\zeta})$, Lemma 4.3 would imply that $\pi$ is cut.)

We now argue that such a jump cannot exist: By assumption it has to meet an open cell of $\nu$ and therefore cannot be fully contained in a vertical feature of $\nu^{(2)}$. So a point at which it enters or leaves $\nu^{(2)}$ must belong to a non-vertical feature of $\nu^{(2)}$ and those are fully contained in $Z(f_{\zeta})$, leading to a contradiction.

To recap, in case (ii) $\pi$ has been cut during the nonrecursive processing of the parent $\zeta$ of $\xi$.

Let us now consider case (i), in which all the triangles of $T_C$ pierce $\sigma_{\xi}$. This is where we do consider the partitioning at $\xi$. If $\xi$ is a leaf, the claim holds because the BSP constructed at $\xi$ eliminates all cycles among the triangles of $T_C$ (in the sense of Lemma 3.1). To make this argument rigorous, we need to bear in mind that we only retain the portions of these cuts within $\sigma_{\xi}$. However, since $\pi$ is fully contained in $\sigma_{\xi}$, any point at which an unclipped cutting segment meets some subpath $\pi_i$ of $\pi$ necessarily lies in $\sigma_{\xi}$, so it lies on the retained clipped portion of that segment.

If $\xi$ is not a leaf, we constructed a partitioning polynomial $f_{\xi}$ for $T_{\xi}$, and used $F_{\xi} = F_{\zeta}f_{\xi}$ to partition $\sigma_{\xi}$ into subcells. If $\pi$ is not fully contained in any subcell of $\sigma_{\xi}$, then it has to meet (and possibly partially overlap) $Z(f_{\xi})$. If $\pi$ meets $Z(f_{\xi})$ at a point or points that lie on one of the subpaths $\pi_i$, then $\pi$ is cut by some trace (in the case of overlap, one or several subarcs of $\pi_i$ might be removed). The only remaining case is that some open vertical jump, say $v_1^−v_2^+$, of $\pi$ meets $Z(f_{\xi})$, and then $\pi$ is cut by Lemma 4.3. Finally, the case where $\pi$ is fully contained in some subcell $\sigma_{\xi'}$ of $\sigma_{\xi}$ is impossible, as it contradicts the definition of $\xi$ as the lowest node satisfying both properties (a) and (b) (namely, that $\pi$ is contained in the cell and that all its triangles pierce the parent cell). Having covered all possible cases, the lemma follows. \qed

This finishes the proof of correctness of our procedure. We still need to fill two gaps: (i) We need an upper bound on $|\Gamma|$, the number of clipped curves that the procedure generates. (ii) We need to cut each triangle into pieces bounded by a constant number of edges and control their number. We now address both of these issues.
4.3 Bounding the Number of Curves

Consider the situation at some recursion node $\xi$, and note that the clipped curves generated at $\xi$ are obtained in two stages. Using an upper bound which is certainly too crude, but suffices for our purposes, we first generate on each triangle of $T$, up to $O(D^3 \log_D^4 n)$ algebraic curves, each of degree at most $O(D^2)$. Then we intersect these curves with the closure of $\sigma_\xi$ to obtain their clipped versions. We start by focusing on the first stage, and define $\chi(T)$, for $T = T_\xi$, to be the maximum number of unclipped curves that our procedure generates on the triangles of $T$, for the fixed choice of $D$ used throughout the recursion. Of course, the actual number of curves depends on the partitioning polynomials constructed throughout the recursion, and $\chi(T)$ maximizes this over all possible choices of partitioning polynomials of degree at most $D$. Put $\chi(n) := \max |T| = n \chi(T)$, where the maximum is taken over all collections $T$ of $n$ non-vertical pairwise disjoint relatively open triangles in $\mathbb{R}^3$. Then $\chi(T)$ satisfies the following recurrence relation (for $|T| > D^2/c$):

$$\chi(T) \leq bD^3 \cdot \chi(c|T|/D^2) + O(|T|D^4 \log_D^4 |T|),$$

where $b$ and $c$ are suitable absolute constants. The overhead term $O(|T|D^4 \log_D^4 |T|)$ comes from a crude bound on the number of wall shadows, which is $O(D^4 \log_D^4 |T|)$ per triangle. This bound dominates the number of all other curves non-recursively constructed on a single triangle at the present node. Maximizing over $T$ produces the recurrence

$$\chi(n) \leq \begin{cases} bD^3 \chi(cn/D^2) + O(nD^4 \log_D^4 n) & \text{for } n > D^2/c, \\ O(D^4) & \text{for } n \leq D^2/c. \end{cases}$$

As is easily verified, the solution of this recurrence is $\chi(n) = O(n^{3/2+\varepsilon})$, for any $\varepsilon > 0$, provided that we choose $D$ so as to satisfy $D^{2\varepsilon} \geq 2bc^{3/2+\varepsilon}$. That is, for a prescribed $\varepsilon$, we need to choose $D = 2^{\Theta(1/\varepsilon)}$, with a suitable constant of proportionality. Conversely, with $D$ specified, that is, with an explicit control over the degree of the curves that we are willing to draw, we have $\varepsilon = O(1/\log D)$.

**Lemma 4.6** Our construction produces $\chi(n) = O(n^{3/2+\varepsilon})$ curves, each of degree $D = 2^{\Theta(1/\varepsilon)}$.

4.4 The Complexity of the Arrangements of the Clipped Curves

To recap, the various steps of the construction generate a collection of curves on the triangles. Altogether $O(D^4 \log_D^4 n)$ curves, each of degree at most $O(D^2)$, are generated for each piercing triangle at each recursive level, the majority of which are the wall shadows. However, for each curve $\gamma$ we retain only its portion within the cell at which $\gamma$ was generated.

Upon termination of the entire recursive process, we take each triangle $\Delta \in T$, and consider the planar map $M_\Delta$ formed on $\Delta$ by the hierarchy of curves constructed
for \( \Delta \). That is, we take each curve \( \gamma \), generated at some recursive node \( \xi \) where \( \Delta \) was a piercing triangle, clip \( \gamma \) to within the cell \( \sigma_\xi \), and draw only the clipped portion \( \gamma \cap \sigma_\xi \); we repeat this operation for all triangles \( \Delta \) and all recursive steps \( \xi \).

Each vertex of \( M_\Delta \) is either (a) an endpoint of a connected component of the clipped portion of some curve \( \gamma \), or (b) an intersection point between two (clipped) curves \( \gamma, \gamma' \), such that either (b.i) both arcs are generated at the same recursive step, within the same cell \( \sigma \), or (b.ii) up to a swap between the arcs, \( \gamma, \gamma' \) are generated within two respective cells \( \sigma, \sigma' \), such that the step that generated \( \sigma' \) is a proper ancestor of the step that generated \( \sigma \). These properties follow easily from the hierarchical nature of our drawings.

The number of clipped connected subarcs, over all the triangles, is at most the number of unclipped curves, which we have shown to be \( O(n^{3/2+\epsilon}) \), with a suitable constant of proportionality, plus the number of cuts that the clipping creates. Any such cut, of some curve \( \gamma \) generated at some step \( \xi \), occurs where \( \gamma \) crosses the boundary of \( \sigma_\xi \), and, in particular, at a point of \( \gamma \cap Z(F_\xi) \); \( \gamma \) may cross several cells into which \( \sigma_\xi \) is split by \( Z(F_\xi) \), but these points are not considered as endpoints of subarcs of \( \gamma \). Since \( \gamma \) is a planar algebraic curve of degree at most \( O(D^2) \), and \( Z(F_\xi) \) is an algebraic surface of degree \( O(D \log_D n) \), it follows from Bézout’s theorem \([14, \text{Thm. 7 in Chap. 8, §7}]\) that the number of pieces into which \( \gamma \) is cut is \( O(D^3 \log_D n) \). The exceptions are endpoints of curves that lie on the edges of the corresponding triangles; we may ignore these vertices, as we have only \( O(D^2) \) such points for each of the curves that we draw, as is easily checked.

We next bound the number of intersection points of clipped arcs with other (clipped) arcs constructed at (proper and improper) ancestral recursive steps. For each arc \( \gamma \), formed along some triangle \( \Delta \), within a cell \( \sigma \) at some recursive step, the number of the ancestral cells of \( \sigma \) is \( O(\log_D n) \), and each of them generates on \( \Delta \) up to \( O(D^4 \log_D^4 n) \) curves of degree at most \( O(D^2) \). For the present argument, treat these curves as drawn in their entirety—this will only increase the number of intersection points on \( \gamma \). Since \( \gamma \) is one of these curves, the number of intersection points of \( \gamma \) with any other curve is \( O(D^4) \), which is a consequence of Bézout’s theorem \([14, \text{Thm. 7 in Chap. 8, §7}]\). It follows that the number of vertices that can be formed along \( \gamma \) is at most \( O(D^8 \log_D^5 n) \), which is quite possibly a gross overestimate, but we do not attempt to optimize it. This slack is indeed quite generous, but it only applies to curves generated at a pair of nodes, one of which is an ancestor of the other. We avoid paying for intersections between curves generated at two unrelated nodes, because each curve is clipped to within the cell of the recursion node at which it is generated.

Adding these two bounds, and multiplying by the number of curves, as provided in Lemma 4.6, we conclude that the overall complexity of the maps \( M_\Delta \), over all triangles \( \Delta \), is

\[
O(D^8 \log_D^5 n) \cdot O(n^{3/2+\epsilon}),
\]

where, as we recall, the prespecified \( \epsilon > 0 \) can be chosen arbitrarily small, and where
\( D = 2^{O(1/\epsilon)} \), with a suitable constant of proportionality. It then follows that, by slightly increasing \( \epsilon \), but keeping it sufficiently small, we can still write the bound as \( O(n^{3/2+\epsilon}) \), with a constant of proportionality of the form \( 2^{O(1/\epsilon)} \).
Remark To optimize the resulting expression, we should set $\varepsilon = c/\sqrt{\log n}$, for a suitable absolute constant $c > 0$, to obtain a bound of the form $n^{3/2} \cdot 2^{O(\sqrt{\log n})}$ on the number of resulting faces. Of course, for this we would have to draw curves of degree $D = 2^{O(\sqrt{\log n})}$. If we instead choose $D = D(\varepsilon)$ not to depend on $n$, this increases the bound on the number of curves and the complexity of their arrangement, but ensures that the curves have “constant” degree (i.e., one that does not depend on $n$) and the pieces produced below are of complexity that does not depend on $n$ but only on $\varepsilon$.

4.5 Final Decomposition into Pseudo-Trapezoids

Finally, we take the planar map $M_\Delta$, for each triangle $\Delta$, and decompose it into regions bounded by at most four edges each, by constructing the trapezoidal decomposition [11] of $M_\Delta$ in some fixed, but arbitrarily chosen “vertical” direction within $\Delta$. Each resulting piece is a pseudo-trapezoid, with at most two vertical sides, and “floor” and “ceiling” sides, each consisting of a monotone subarc of one of the curves we have drawn on $\Delta$ (or part of an edge of $\Delta$), and thus having degree at most $O(D^2)$.

The number of trapezoids is proportional to the complexity of $M_\Delta$, which in turn is proportional to the number of its vertices, plus the number of points at which the drawn curves have vertical tangents. As the curves have degree $O(D^2)$, each curve can have at most $O(D^4)$ such tangency points (by Bézout’s theorem [14, Thm. 7 in Chap. 8, §7] applied to $g$ and $\partial g/\partial y$, if the local coordinates within $\Delta$ are $x$ and $y$, $y$ vertical, and the curve is defined by $g(x, y) = 0$ for a square-free polynomial $g$). Using the analysis from the preceding subsection, this brings us to the main result of the paper.

Theorem 4.7 Let $\mathcal{T}$ be a collection of $n$ pairwise disjoint non-vertical relatively open triangles in $\mathbb{R}^3$. Then, there is a subdivision of the triangles of $\mathcal{T}$ into relatively open pseudo-trapezoids so that the depth relation among them is acyclic.

(a) One such subdivision contains $n^{3/2} \cdot 2^{O(\sqrt{\log n})}$ pseudo-trapezoids, each bounded by at most two arcs of maximum degree $2^{O(\sqrt{\log n})}$ and at most two line segments.

(b) Alternatively, for any given $\varepsilon > 0$, there is a subdivision consisting of $O(n^{3/2+\varepsilon})$ pseudo-trapezoids bounded by at most two arcs of degree that depends only on $\varepsilon$ and at most two line segments; here the implied constants of proportionality in the bound on the number of pieces and in the degree bound, are of the form $2^\Theta(1/\varepsilon)$.

5 Extensions and Generalizations

Allowing Vertical Triangles We can extend our result to handle vertical triangles by the following observation: Each vertical triangle is replaced by its three edges, each of which is treated as a degenerate zero-area triangle. It is easy to check that our construction goes through essentially unchanged, eliminating depth cycles in the resulting family of real and degenerate triangles. Since the segments are one-dimensional, cuts along them are removals of isolated points. We then cut the original triangle along the vertical segments passing through such cut points. We omit the easy details.
An Algorithm  Combining case (b) of Theorem 4.7 with Proposition 2.2 yields an $O(n^{3/2+\varepsilon})$ expected time algorithm to cut $n$ triangles in $O(n^{3/2+\varepsilon})$ pseudo-trapezoids as described in the theorem. For more details about efficient implementation of the various steps of the algorithm, see Agarwal et al. [1, Thm. 9].

Theorem 5.1  Let $T$ be a collection of $n$ pairwise disjoint non-vertical relatively open triangles in $\mathbb{R}^3$ and let $\varepsilon > 0$. Then, we can cut the triangles of $T$ into $O(n^{3/2+\varepsilon})$ pseudo-trapezoids so that the depth relation among them is acyclic as in Theorem 4.7 (b), in expected $O(n^{3/2+\varepsilon})$ time.

Remark  Notice that the running time of the algorithm in Proposition 2.2 depends heavily on the degree $D$. In particular, using it to construct the slightly better partition of Theorem 4.7 (a) produces a superpolynomial-time algorithm. At this point, we do not know how to construct it in polynomial time.

Higher Degree Curves and Surfaces  Suppose that, instead of a set of $n$ triangles, our input consists of a set $\Sigma$ of $n$ pairwise disjoint non-vertical $xy$-monotone surface patches cut out of algebraic surfaces of degree bounded by a constant $b > 0$. (The patches are $xy$-monotone in the sense that each vertical line intersects each of them in at most one point. Patches intersecting vertical lines in several points can be handled by splitting them into monotone subpatches. For simplicity, we do not allow the patches to contain a vertical segment of non-zero length.) We assume that the boundary of each surface is formed by at most $b$ arcs of curves of degree at most $b$.

The notion of a depth relation extends naturally to this situation, with one minor exception: It is now possible for two patches $A$ and $B$ to satisfy $A \succ B$ and $A \prec B$ simultaneously, i.e., to form a cycle of length two in the depth relation.

A careful examination of our analysis reveals that both the combinatorial bound and the algorithm apply to this more general situation, essentially verbatim. Therefore we have:

Theorem 5.2  Let $\Sigma$ be a collection of $n$ pairwise disjoint non-vertical relatively open surface patches in $\mathbb{R}^3$, each contained in an algebraic surface of degree at most $b$, so that the boundary of each patch is formed by at most $b$ algebraic curves of degree at most $b$. Then, we can cut the patches of $\Sigma$ into relatively open pairwise disjoint pseudo-trapezoids so that the depth relation among these pseudo-trapezoids is acyclic.

(a) We can construct $n^{3/2}2^{O(\sqrt{\log n})}$ pseudo-trapezoids, each bounded by at most two arcs of maximum degree $2^{O(\sqrt{\log n})}$ and at most two arcs of degree at most $b$.

(b) Alternatively, we can obtain, for any given $\varepsilon > 0$, $O(n^{3/2+\varepsilon})$ pseudo-trapezoids, each bounded by at most two arcs of degree that depends only on $\varepsilon$ and at most two arcs of degree at most $b$; here the implied constants of proportionality in the bound on the number of pieces and in the degree bound, are of the form $2^{O(1/\varepsilon)}$.

A randomized algorithm can construct the pieces in expected $O(n^{3/2+\varepsilon})$ time, if the patches are in general position.

All implied constants depend on $b$.  

Springer
6 Discussion

In this paper we have made substantial progress on the long-standing problem of eliminating depth cycles in a set of pairwise openly disjoint triangles in $\mathbb{R}^3$. On the positive side, our solution is almost optimal in the worst case, in terms of the number of pieces, as this number is only slightly larger than the $\Omega(n^{3/2})$ worst-case lower bound noted in [13]. However, a notable disadvantage of our solution is that the cuts are by constant-degree algebraic arcs, rather than, ideally, by straight segments. Is there a way, perhaps by combining our methods with those of de Berg [10], to produce a partition of triangles into triangular pieces, in subquadratic time? See further discussion in the introduction.

One direction for future research is to further tighten the bound. In fact, at this point there is no evidence that the correct answer is not simply $\Theta(n^{3/2})$, even for the much simpler case of lines.

As noted above, the BSP partition of [21] has the stronger property that the depth relation of the resulting pieces is acyclic with respect to any viewing point or direction. Our solution does not seem to have this property, so a natural question is whether one can cut the triangles into a subquadratic number of simple pieces that have this stronger property, or whether $\Omega(n^2)$ pieces are required, in the worst case, in order to have the depth order of the resulting pieces be free of cycles, for every viewing direction (and possibly also for every perspective view).

Acknowledgements The authors wish to express their gratitude to Saugata Basu for his help in discussing matters of algebra, and to Saugata Basu, Sylvain Cappell, Jeff Erickson, and Josh Zahl for some assistance in matters of topology. The presentation of the current version of this paper was much improved thanks to the comments of the anonymous referees.

References

1. Agarwal, P.K., Aronov, B., Ezra, E., Zahl, J.: An efficient algorithm for generalized polynomial partitioning and its applications. In: 35th International Symposium on Computational Geometry. Leibniz Int. Proc. Inform., vol. 129, # 5. Leibniz-Zent. Inform., Wadern (2019)
2. Agarwal, P.K., Grove, E.F., Murali, T.M., Vitter, J.S.: Binary space partitions for fat rectangles. SIAM J. Comput. 29(5), 1422–1448 (2000)
3. Aronov, B., de Berg, M., Gray, C., Mumford, E.: Cutting cycles of rods in space: hardness and approximation. In: 19th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1241–1248. ACM, New York (2008)
4. Aronov, B., Ezra, E., Zahl, J.: Constructive polynomial partitioning for algebraic curves in $\mathbb{R}^3$ with applications. In: 30th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 2636–2648. SIAM, Philadelphia (2019). Updated version at arXiv:1904.09526
5. Aronov, B., Koltun, V., Sharir, M.: Cutting triangular cycles of lines in space. Discrete Comput. Geom. 33(2), 231–247 (2005)
6. Aronov, B., Miller, E.Y., Sharir, M.: Eliminating depth cycles among triangles in three dimensions. In: 28th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 2476–2494. SIAM, Philadelphia (2017)
7. Aronov, B., Sharir, M.: Almost tight bounds for eliminating depth cycles in three dimensions. Discrete Comput. Geom. 59(3), 725–741 (2018)
8. de Berg, M.: Ray Shooting, Depth Orders and Hidden Surface Removal. Lecture Notes in Computer Science, vol. 703. Springer, Berlin (1993)
9. de Berg, M.: Linear size binary space partitions for uncluttered scenes. Algorithmica 28(3), 353–366 (2000)
10. de Berg, M.: Removing depth-order cycles among triangles: an algorithm generating triangular fragments. Discrete Comput. Geom. (2019). https://doi.org/10.1007/s00454-019-00102-0
11. de Berg, M., Cheong, O., van Kreveld, M., Overmars, M.: Computational Geometry. Algorithms and Applications. Springer, Berlin (2008)
12. de Berg, M., Overmars, M., Schwarzkopf, O.: Computing and verifying depth orders. SIAM J. Comput. 23(2), 437–446 (1994)
13. Chazelle, B., Edelsbrunner, H., Guibas, L.J., Pollack, R., Seidel, R., Sharir, M., Snoeyink, J.: Counting and cutting cycles of lines and rods in space. Comput. Geom. 1(6), 305–323 (1992)
14. Cox, D., Little, J., O’Shea, D.: Ideals, Varieties, and Algorithms. An Introduction to Computational Algebraic Geometry and Commutative Algebra. Undergraduate Texts in Mathematics. Springer, New York (2007)
15. Dumitrescu, A., Tóth, C.D.: Binary space partitions. In: Encyclopedia of Algorithms, pp. 220–223. Springer, Boston (2016)
16. Fulton, W.: Introduction to Intersection Theory in Algebraic Geometry. CBMS Regional Conference Series in Mathematics, vol. 54. American Mathematical Society, Providence (1984)
17. Guth, L.: Polynomial partitioning for a set of varieties. Math. Proc. Camb. Philos. Soc. 159(3), 459–469 (2015)
18. Guth, L., Katz, N.H.: On the Erdős distinct distances problem in the plane. Ann. Math. 181(1), 155–190 (2015)
19. Har-Peled, S., Sharir, M.: Online point location in planar arrangements and its applications. Discrete Comput. Geom. 26(1), 19–40 (2001)
20. Harris, J.: Algebraic Geometry. A First Course. Graduate Texts in Mathematics, vol. 133. Springer, New York (1992)
21. Paterson, M.S., Yao, F.F.: Efficient binary space partitions for hidden-surface removal and solid modeling. Discrete Comput. Geom. 5(5), 485–503 (1990)
22. Sharir, M., Agarwal, P.K.: Davenport–Schinzel Sequences and Their Geometric Applications. Cambridge University Press, Cambridge (1995)
23. Solan, A.: Cutting cycles of rods in space. In: 14th Annual ACM Symposium on Computational Geometry, pp. 135–142. ACM, New York (1998)
24. Spanier, E.H.: Algebraic Topology. McGraw-Hill, New York (1966)
25. Tóth, C.D.: Binary space partitions: recent developments. In: Combinatorial and Computational Geometry. Mathematical Sciences Research Institute Publications, vol. 52, pp. 529–556. Cambridge University Press, Cambridge (2005)
26. Tóth, C.D.: Binary space partitions for axis-aligned fat rectangles. SIAM J. Comput. 38(1), 429–447 (2008)
27. Warren, H.E.: Lower bounds for approximation by nonlinear manifolds. Trans. Am. Math. Soc. 133, 167–178 (1968)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Boris Aronov1 · Edward Y. Miller2 · Micha Sharir3

Boris Aronov
boris.aronov@nyu.edu

Edward Y. Miller
emiller@cims.nyu.edu

Micha Sharir
michas@tauex.tau.ac.il
Department of Computer Science and Engineering, Tandon School of Engineering, New York University, Brooklyn, NY 11201, USA

Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, USA

Blavatnik School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel