COMBINATORICA
Akadémiai Kiadó – Springer-Verlag

COMBINATORICA 10 (2) (1990) 137–173

TRIANGLES IN SPACE OR BUILDING (AND ANALYZING) CASTLES IN THE AIR

B. ARONOV and M. SHARIR*

Received November 1, 1988
Revised February 27, 1989

We show that the total combinatorial complexity of all non-convex cells in an arrangement of n (possibly intersecting) triangles in 3-space is $O(n^{7/3}\log n)$ and that this bound is almost tight in the worst case. Our bound significantly improves a previous nearly cubic bound of Pach and Sharir. We also present a (nearly) worst-case optimal randomized algorithm for calculating a single cell of the arrangement and an alternative less efficient, but still subcubic algorithm for calculating all non-convex cells, analyze some special cases of the problem where improved bounds (and faster algorithms) can be obtained, and describe applications of our results to translational motion planning for polyhedra in 3-space.

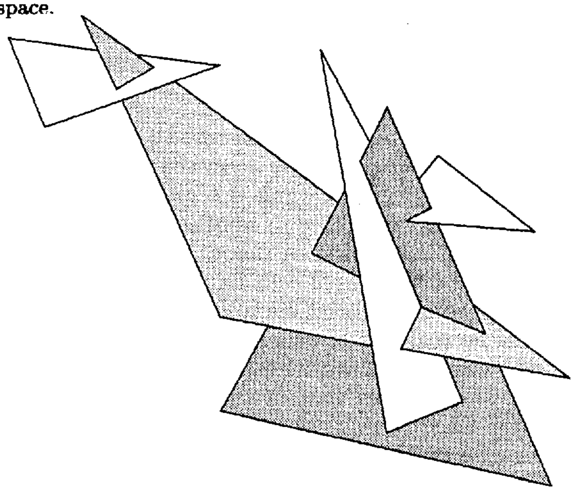

Fig. 1. An arrangement of triangles in $\mathbf{R}^3$

*Work on this paper by the first author has been supported by an AT&T Bell Laboratories PhD Scholarship. Work on this paper by the second author has been supported by Office of Naval Research Grant N00014-87-K-0129, by National Science Foundation Grant No. NSF-DCR-83-20085, by grants from the Digital Equipment Corporation, the IBM Corporation, and by a research grant from the NCRD — the Israeli National Council for Research and Development. A preliminary version of this paper has appeared in *Proc. 4th ACM Symp. on Computational Geometry,* Urbana, Illinois, 1988.

AMS subject classification (1980): 51–04, 52 A 37, 68 R 05.

1. Introduction

1.1. Terminology

Consider a collection G of n (possibly intersecting) closed flat triangular (or, more generally, arbitrary convex) objects $\Delta_1, \ldots, \Delta_n$ in three-dimensional Euclidean space $\mathbb{R}^3$ (See Fig. 1.). Let $A = A(G)$ denote the *arrangement* of these triangles, i.e., the subdivision of $\mathbb{R}^3$ induced by them; thus A is a decomposition of 3-space into pairwise disjoint connected cells of 0,1, 2, or 3 dimensions, where

1. a 3-dimensional cell is a connected component of $\mathbb{R}^3 - \bigcup_i \Delta_i$,
2. a 2-dimensional cell (a *face*) is a connected component of $int(\Delta_i) - \bigcup_{j \neq i} \Delta_j$ for some i (where $int(\Delta)$ denotes the relative interior of Δ),
3. a 1-dimensional cell (an *edge*) is either a maximal connected portion of (the relative interior of) an edge of some triangle Δ_i, which does not intersect any other triangle, or a maximal connected portion of (the relative interior of) the intersection segment $\Delta_i \cap \Delta_j$ of two triangles, which does not meet a third triangle, and
4. a 0-dimensional cell (a *vertex*) is a vertex of a triangle, an intersection of an edge of one triangle with another triangle, or an intersection point of three distinct triangles.

To simplify the analysis, we henceforth assume that our triangles are placed in *general position* in space, meaning that no four share a point, no three intersect in more than a point, no two in more than a segment, no two edges of different triangles intersect, no triangle vertex lies in another triangle, and no triangle edge intersects another triangle in more than a point. These assumptions guarantee in particular that each of the above types of cells have the correct dimensionality. Since our goal is to analyze the combinatorial complexity of (certain portions of) $A(G)$, this assumption involves no real loss of generality, because any degenerate layout of triangles can always be viewed as a limiting case of layouts in general position, whose combinatorial complexity dominates that of the limit configuration. Roughly, this can be argued as follows: Each triangle in the arrangement can be slightly enlarged to eliminate all degeneracies involving triangle boundaries. Moreover, this expansion can be performed without changing the topology, or reducing the number of faces, on any of the cell boundaries. The remaining degeneracies can be eliminated by a slight perturbation of the planes containing the triangles — such a perturbation never destroys a face, though it may create new faces. Thus we obtain an arrangement of triangles in general position in which no cell is bounded by fewer faces than it was in the original arrangement. Theorem A.4. of Appendix A, to be discussed below, essentially states that the complexity of any set of cells is dominated by the number of faces on their combinatorical boundaries, implying our claim.

We will use the following additional terminology. A vertex of a triangle will be called a *corner*, an edge of a triangle will be called an *exposed edge*, and the 1-dimensional cells of $A(G)$ into which it is decomposed will be called *exposed segments*. Finally, the unqualified term *cell* will be used to denote a 3-dimensional cell of $A(G)$.

At various points in our analysis we will have to consider *planar arrangements of segments* which are, similar to the 3-dimensional case, partitions of a plane into *faces, edges,* and *vertices* induced by a collection of segments in that plane. As a bounded face of such an arrangement need not be simply connected, its boundary will in general consist of a single *outer component* that encloses the entire face and zero or more *islands* each enclosing a portion of the plane that does not belong to the face. An unbounded face has a similar structure, except its boundary does not have an outer component.

1.2. Motivation and Main Results

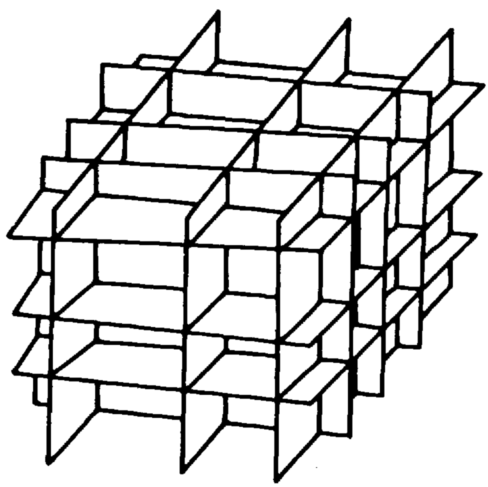

Fig. 2. A grid-like arrangement

Our goal is to obtain sharp upper and lower bounds on the maximum (worst-case) *combinatorial complexity* of any single cell C of $A(G)$, that is the number of vertices, edges, and faces of $A(G)$ lying along the boundary of C. Consider first the following classification. A cell C of $A(G)$ is *interesting* if its closure $\bar{C}$ contains at least one exposed segment (it follows easily from the definition that, if $\bar{C}$ intersects an exposed segment, it fully contains it). All other cells of $A(G)$ are called *dull*. It is easily checked that a dull cell of $A(G)$ is (the interior of) a convex polyhedron bounded by (some of) the planes containing the triangles; its combinatorial complexity is thus only $O(n)$. Interesting cells, on the other hand, may have highly irregular shape; in fact, if n is sufficiently large, an interesting cell may approximate any open connected semi-algebraic set in $\mathbb{R}^3$, its boundary may have arbitrary genus, etc.* Let $\zeta(n)$ be the maximum combinatorial complexity of a single (interesting) cell, over all possible arrangements of n triangles in general position. Simple examples show that $\zeta(n)$ is $\Omega(n^2)$ (see Fig. 2). A more sophisticated example demonstrates that $\zeta(n)$

*Note that, although in the simple case of triangles (or arbitrary *flat* objects) the notions of dull and interesting coincide with those of convex and non-convex, respectively, this is not necessarily the case for more general surface patches.

is $\Omega(n^2\alpha(n))$ [23], where $\alpha(n)$ is the extremely slowly growing inverse Ackermann function. In fact, the combinatorial complexity of the *upper envelope* of the triangles in G (which is a portion of the boundary of the unbounded cell of $A(G)$, certainly an interesting cell), is always $O(n^2\alpha(n))$ [23]. Following the results of [23], as well as the work of [24] and [12], mentioned below, on the two-dimensional analog of our problem, it is natural to conjecture that

$$\zeta(n) = \Theta(n^2\alpha(n)). \qquad (Conjecture)$$

We have been able to prove this conjecture only for a few special cases, and in general it is still an open problem. The only previously known non-trivial upper bound on $\zeta(n)$, given in [23], is $O(n^{3-1/49})$; it is proved using a fairly intricate analysis based on extremal hyper-graph theory. (It is easy, by the way, to show that the worst-case combinatorial complexity of the *full* arrangement $A(G)$ is $\Theta(n^3)$.) Our results (stated below) improve and generalize this bound, and we believe that our techniques can be extended to yield a bound close to that conjectured above.

The main motivation (also discussed in [13,23]) for studying this problem comes from motion planning. Specifically, let B be an arbitrary polyhedral object bounded by K faces, edges, and vertices and free to *translate* in three-dimensional space while avoiding stationary polyhedral obstacles bounded by a total of n faces, edges, and vertices. Using the standard technique of expanding obstacles in configuration space, it is easily seen (cf. [13,23]) that the 3-dimensional space FP of all free placements of B can be represented as a union of certain (3-dimensional) cells of $A(G)$, for the collection G of $O(Kn)$ triangles obtained from the Minkowski (vector) differences between faces, edges, and vertices of the obstacles and vertices, edges, and faces of B, respectively. Moreover, given an initial placement z_0 of B, it suffices to calculate only the connected component of FP containing z_0, because no other portion of FP can be reached from z_0 by a collision-free motion. Thus, analysis of the combinatorial complexity and efficient calculation of a single cell of $A(G)$ are major components in the design and analysis of efficient algorithms for this translational motion planning problem.

Before presenting our results, let us first review the analogous (and relatively simpler) situation in two dimensions, namely the case of an arrangement A of n line segments in the plane. It is known that the worst-case combinatorial complexity of a single face of A is $\Theta(n\alpha(n))$, which is asymptotically the same as the worst-case complexity of the upper envelope of n segments [18,24,28]. Also, the worst-case combinatorial complexity of all interesting (i.e., non-convex) faces of A is $O(n^{4/3})$ [1] (see also [12]); this bound is almost tight, as the complexity in question can be $\Omega(n^{4/3})$ [15]. The above upper bound is a special case of the result of [1], who argue that the complexity of any m distinct faces in a planar arrangement of n segments is

$$O(m^{2/3}n^{2/3} + n\log n).$$

It was further shown in [1] that the bound can in fact be generalized to

$$O(m^{2/3}p^{1/3} + n\log n),$$

where p is the number of pairs of intersecting segments. (Note that, for $m = \Theta(p)$, it yields $O(p + n\log n)$, which nearly coincides with the actual $\Theta(p + n)$ complexity of the entire arrangement.) Furthermore, if the m faces are those intersected by another (fixed) segment, their total complexity is only $O(n\alpha(n))$, as has been shown in [10]. These results for two-dimensional arrangements are closely related to bounds that we obtain and are extensively used in our analysis. Moreover, they clearly point toward the conjecture made above.

As already mentioned, we have not been able to obtain a general near-quadratic upper bound for $\zeta(n)$. Instead, we consider here the problem of analyzing a larger quantity — the maximum total combinatorial complexity $\Phi(n)$ of *all* interesting cells in any arrangement of n triangles in 3-space. Our main result is:

Theorem 5(c). $\Phi(n) = O(n^{7/3}\log n)$; *this is almost tight since* $\Phi(n) = \Omega(n^{7/3})$.

We thus obtain a significant improvement over the bound of [23], both because our bound is sharper and because it applies to the total complexity of all interesting cells, rather than that of a single one. In fact, we obtain a somewhat more general bound that depends on the number p of pairs and the number t of triples of intersecting triangles in G. Specifically, in Theorem 5(a) we show that, if $t \geq p\log^3 n$ and $p \geq n$, the total complexity of the interesting cells in an arrangement of n triangles is

$$O(p^{2/3}t^{1/3}\log n),$$

which reduces to the previous bound by putting $p = O(n^2)$ and $t = O(n^3)$. Moreover, our analysis can be extended to obtain similar bounds on $\Phi(n)$ for arrangements of arbitrary convex flat objects (called *convex plates* in [23]), with the possible addition of a linear correction term to accommodate the complexity of the "exposed" boundary of these plates.

The results outlined above generalize the $O(n^{7/3}\log n)$ upper bound of [11] on the complexity of $O(n^2)$ cells in an arrangement of n *planes*. In fact, our Theorem 3 with $t = O(n^3)$, $p = O(n^2)$ yields a generalization of the bound

$$O(m^{2/3}n\log n + n^2),$$

given in [11] for the complexity of any m cells in a plane arrangement, to arrangements of triangles. However, our result is valid only if all non-convex cells are included in the collection of desired cells, so in our case m has to be "large", i.e. at least $3n + 2p$ (see below for details and explanation of this term).

We also consider the task of actual calculation of a single cell of $A(G)$ and present a randomized algorithm, whose expected running time is $O(\zeta(n)n^{\delta})$ for any $\delta > 0$, with the constant of proportionality depending on δ. Moreover, we demonstrate that the same algorithm is faster when applied to some special classes of arrangements. We also describe an alternative algorithm that computes all interesting cells in an arrangement of n triangles in expected time $O(n^{8/3}\log^{14/3} n)$ and $O(n^{7/3}\log n)$ space, and present a modification of this algorithm that computes any subset of cells in the arrangement. For details, refer to Section 5.2.

There are two special cases where both complexity bounds and algorithmic performance can be significantly improved. One case involves triangles lying in planes having only a small number f of orientations (e.g., when each triangle is parallel to one of the three coordinate planes). In this case we show that the maximum complexity of a single cell as well as that of all interesting cells is $\Theta(n^2)$ for a fixed f.

More precisely, the latter complexity is $\Theta(fn^2)$ for $f = O(n^{1/3})$ and the former is $\Omega(n^2\alpha(f))$. We also provide an efficient $O((M + n^2)f \log n)$-time *deterministic* algorithm for the calculation of any subset of cells in such an arrangement, given a point in each cell, where M is the total complexity of the cell(s) being computed.

The second restricted class of arrangements contains only two types of objects: (1) arbitrary horizontal convex plates (parallel to the xy-plane) and (2) vertical rectangles (whose planes are parallel to the z-axis and whose top and bottom sides are parallel to the xy-plane). In this case we show that the maximum complexity of a single cell is $\Theta(n^2\alpha(n))$.

In addition, we argue that our unmodified general algorithm computes any single cell of an arrangement of n triangles, either with only a constant number of orientations or in the second special case above, in expected time $O(n^{2+\delta})$ (for any $\delta > 0$).

The technique used to show the $O(n^{7/3} \log n)$ upper bound on $\Phi(n)$ is relatively simple. The proof is based on the results concerning the combinatorial complexity of many faces in an arrangement of segments in the plane, summarized above. In addition, we adapt and extend some of the technical tools developed in [11,12] for studying related problems. More specifically, supposing that bounds on the complexity of all interesting cells in two subarrangements have been obtained, we seek to establish a sharp relationship between these complexities and the complexity of all interesting cells in the arrangement formed by the union (i.e., overlay) of these two subarrangements. Such relationships, called "combination lemmas" have been derived and exploited in [1,12] for the case of lines or segments in the plane and in [11] for the case of planes in 3-space. We obtain a similar relationship for arrangements of triangles (see Lemma 1 below), using a proof technique that is somewhat different from (and, we believe, simpler than) those employed in [11,12].

To analyze the time performance of our main algorithm we prove an important technical result ("the Slicing Theorem") which may be of independent interest. It states roughly that any collection of cells in an arrangement of triangles can be subdivided into tetrahedra without substantially increasing the total complexity of this collection.

The paper is organized as follows: In section 2 we obtain our main results on the combinatorial complexity of all interesting cells. Section 3 derives the Slicing Theorem. Section 4 discusses the special cases mentioned above and demonstrates the improved bounds. Algorithms for computing single and multiple cells in arrangements of triangles are presented in Section 5, and a discussion of some applications and open problems is given in Section 6. Appendix A contains some basic facts, needed for our analysis, concerning the topology of cells in a three-dimensional arrangement of triangles, Appendix B exemplifies the usefulness of our techniques by presenting a simple alternative proof of (a variant of) the 2-dimensional combination lemma for polygonal regions used in [12].

2. The Complexity of All Interesting Cells

Let $G = \{\Delta_1, \ldots, \Delta_n\}$ be a collection of triangles in 3-space as in the introduction. We determine the total complexity of the interesting cells of $A(G)$ by analyzing a more general problem:

Consider collections G of n triangles in $\mathbb{R}^3$ with at most p intersecting pairs and at most t intersecting triples, and any set P of $m \geq 2p + 3n$ points, *containing at least one point on each exposed segment* of $A(G)$; we wish to determine the maximum complexity $C(m, n, p, t)$ of all cells of $A(G)$ that contain at least one of the given points in their interior or on their boundary, where the complexity of a cell containing multiple points is to be counted *only once*, and where the maximum is taken over all such collections G and sets of points P. (Our points will be chosen so that each point lies in the closure of a unique cell.)

Note first that the number of exposed segments in $A(G)$ is $2p+3n \leq 2\binom{n}{2}+3n = O(n^2)$. Indeed, each endpoint of an exposed segment is either a triangle corner or an endpoint of the intersection segment of a pair of triangles. If P contains only points lying on exposed segments, one point on each segment, the *marked* cells of $A(G)$ (i.e., the cells whose closures contain those points) are precisely all the interesting cells. Hence, $C(2p + 3n, n, p, t)$ serves as an upper bound on the complexity of all interesting cells of $A(G)$, and $C\left(2\binom{n}{2} + 3n, n, \binom{n}{2}, \binom{n}{3}\right)$ is an upper bound on $\Phi(n)$ (and, therefore, also on $\zeta(n)$). In the more general problem formulation, we consider all interesting cells plus possibly some additional dull cells.

We employ the following divide-and-conquer approach to analyzing $C(m, n, p, t)$. Partition G into two subsets G_1 and G_2 with $|G_1| = \lfloor n/2 \rfloor, |G_2| = \lceil n/2 \rceil$. Refer to the elements of G_1 as *red triangles*, and to those of G_2 as *blue*. Obtain recursively the "red" marked cells $R_1, \ldots, R_r$ of $A(G_1)$, and the "blue" marked cells $B_1, \ldots, B_b$ of $A(G_2)$ (so that the closure of each R_i and each B_j contains at least one point of P). For each point $p_i \in P$, let R_{r_i}, B_{b_i} be the red and the blue cells marked by p_i, respectively. Let E_i be the connected component of $R_{r_i} \cap B_{b_i}$ containing p_i, i.e., the cell of $A(G)$ marked by p_i; we refer to the cells E_i as *purple cells*. Let the total complexity of all red cells be ρ and that of all blue cells be β. Our goal is to show that the total complexity of the purple cells is at most

$$\rho + \beta + O(m^{2/3}t^{1/3} + p \log p).$$

We will refer to this property, to be proved below, as the *Combination Lemma for an Arrangement of Triangles.* It generalizes a similar combination lemma obtained in [11] for planes in $\mathbb{R}^3$.

2.1. The Combination Lemma

Before proceeding with the analysis, let us note that the complexity of a cell C is simply the sum of the number of vertices, edges, and faces of the boundary ∂C of C. However, in order to simplify the representation of cell boundaries for our analysis, we will count some of these features with multiplicities. Namely, imagine replacing each triangle Δ by a "puffy triangle" Δ^* which is a thin body bounded by two copies of Δ, slightly curved outward from Δ and seamed together at the (relative) boundary of Δ. If the thickness of these puffy triangles is kept sufficiently small, then our general position assumption on the triangles in G guarantees that the resulting arrangement of the puffy triangles (more precisely, the common exterior of these thin bodies) maintains the combinatorial and topological structure of $A(G)$.

except that each face of $A(G)$ now appears as two distinct faces, one on each side of the corresponding puffy triangle and, similarly, each (intersection) edge of $A(G)$ appears as four distinct edges, each (triple-intersection) vertex of $A(G)$ appears as eight distinct vertices, etc. In the subsequent analysis, we will follow this "puffy model" and take the resulting multiplicities of boundary features into account.

In Appendix A we also show that the number of edges and vertices on the boundary ∂C of a cell C is bounded by a linear function of the number of faces of ∂C plus a certain correction term which accounts for the fact that the vertices, edges, and faces along ∂C do not necessarily constitute a triangulation of that boundary and that C and/or some of its faces need not be simply connected. It is demonstrated as well that the total sum of these correction terms, over all cells in the arrangement, is only $O(n+p)$. Thus the complexity of all interesting cells can be estimated by simply bounding the total number of faces on the boundaries of such cells, and adding an $O(n+p)$ "overhead" term, which will be absorbed anyway in the bound that we will obtain. In the remainder of our discussion we will use the term "complexity of a cell" to refer exclusively to the number of its faces.

Let us now return to the partitioning of G into red and blue subcollections G_1, G_2, as above. Consider a resulting purple cell E; it is a connected component of the intersection of a red cell R and a blue cell B. Each face of E is a portion of a (red) face of R or of a (blue) face of B. We will separately bound the number of blue faces and the number of red faces of E (and sum these bounds over all purple cells). Consider red faces first. Notice that the same face of R can yield a number of purple faces that may appear in E as well as in other purple cells. However, creation of *one* purple face out of each red face is already accounted for by the overall "red" complexity ρ, so let us concentrate on the number of "extra" faces cut out of red faces — these appear either because a portion of the boundary of some blue cell B may split a red face f into a number of disconnected pieces, or because there may be two points of P in (the closure of) R belonging to different purple cells and each of these cells may have a portion of f on its boundary. The former situation occurs only if at least one of B, f is non-convex.

We start with the red cells (i.e., those marked by P) in $A(G_1)$, and proceed to construct the final purple cells incrementally. Each step of this procedure involves adding a blue triangle $\Delta = \Delta_i$ to the subarrangement A_{i-1} of all the red and the first $i-1$ blue triangles, to obtain the next subarrangement A_i (thus $A_0 = A(G_1)$ and $A_{\lceil n/2 \rceil} = A(G)$). A "currently purple" cell $E^{(i-1)}$ (i.e., a cell of A_{i-1} marked by P) may be modified by the addition of Δ in one of several ways — it may be trimmed but remain a single cell with the same topological structure, it may remain a single cell but be cut by Δ in a way which changes the topology of its boundary, it may be split into two or more subcells, each marked by a point of P, or its boundary may acquire a new connected component (if Δ is entirely contained in $E^{(i-1)}$). (We need not be concerned with the last case though, as in such a situation the introduction of Δ has no influence on the number of red faces in purple cells.) We wish to estimate the number of *additional* red faces created on the boundary of currently purple cells (in the arrangement A_i) by the introduction of Δ. Summing the resulting bounds over all Δ_i, and repeating the argument for blue faces (starting with the blue arrangement and adding to it red triangles one at a time), we will obtain an upper bound for the purple complexity in terms of the red and blue complexities.

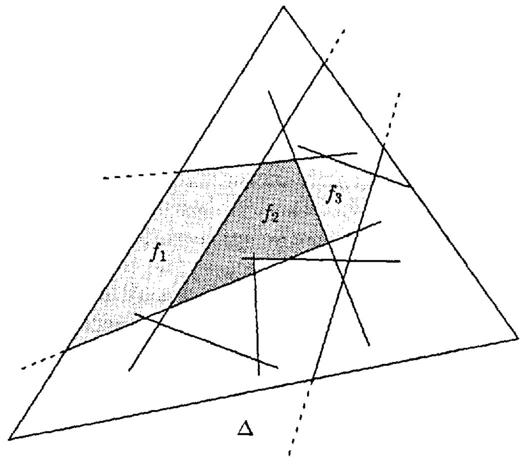

Fig. 3. The arrangement $Q(\Delta)$ with three "purple" faces

Consider the intersection of Δ with the current purple arrangement A_{i-1}. It appears as an arrangement $Q = Q(\Delta)$ of segments in the plane of Δ (all clipped to within Δ), each face of which corresponds to a face of some cell (actually to a pair of faces — one on each side of Δ) in the updated arrangement A_i. For convenience, the relative boundary $\partial\Delta$ of Δ is also added to Q. Let p_i be the number of segments (other than the three edges of Δ) participating in Q (in other words, the number of triangles Δ_j intersecting Δ for $j < i$). Let t_i be the number of intersecting pairs of segments of Q (again, we do not include intersections involving $\partial\Delta$). As discussed in the introduction, a bounded face of Q need not be simply connected; its boundary consists of one exterior connected component and of zero or more interior components — so called *islands*.

We will refer to a face of Q incident to $\partial\Delta$ as a "boundary face". If we erase the three edges of Δ, the boundary faces fuse into the single unbounded cell of the arrangement formed solely by the intersections of triangles Δ_j (with $j < i$) with Δ, so that their total complexity, by [24], is at most $O(p_i\alpha(p_i))$. (See Fig. 3.)

Observe that a "boundary face" f (such as face f_1 in Fig. 3) cannot split a currently purple cell E of A_{i-1} into two subcells, as f does not completely cut through E. Nevertheless, it can produce additional red faces. In fact, every segment e on the boundary of f that does not touch $\partial\Delta$ represents a cut of some purple face f' of A_{i-1} (note that f' need not be a red face, but only an earlier added blue face; also, f' need not be split in two by e, but just have e reduce the number of its islands). In other words, the total number of additional red faces produced by all boundary faces on Δ is no larger than the total complexity of all those boundary faces, i.e. $O(p_i\alpha(p_i))$.

Let us now consider an internal face f of $Q(\Delta)$ (such as faces f_2, f_3 in Fig. 3). Recall that we restrict our attention to faces f which are contained in a currently purple cell. Since f is internal, it must either split such a cell E into two subcells E_1, E_2 or change the topology of the boundary of E without splitting E. In the former case, if one of the two resulting subcells (say E_1) does *not* contain points of

P, no additional red faces are produced because, by our choice of P, E_1 must be dull, i.e. convex (for otherwise its closure would have to contain an exposed segment, and thus also a point of P). In particular, f itself is convex and thus cuts each face of E at most once. Hence each face of E yields at most one face of E_2, as asserted. If, on the other hand, both E_1 and E_2 contain points of P, the number of new red faces created is at most proportional to the complexity of f (by an argument analogous to the case of a boundary face). Let the number of such internal "splitting" faces of $Q(\Delta_i)$ be s_i. Finally, there is the case when introduction of an internal face f does not split E into two subcells, but only changes the topology of its boundary. Again the number of additional red faces being created is at most the number of edges of f. Let the number of such "cutting-but-not-splitting" faces of $Q(\Delta_i)$ be c_i. Hence the total number of additional red faces created by Δ_i is $O(p_i\alpha(p_i))$ plus the complexity of $s_i + c_i$ faces in a planar arrangement of p_i segments of which t_i pairs actually intersect. The latter complexity, by [1], is

$$O((s_i + c_i)^{2/3}t_i^{1/3} + p_i \log p_i). \tag{1}$$

Before proceeding with the analysis, let us observe that $\sum_{i=1}^{\lceil n/2 \rceil} p_i \le p$ since in the incremental construction the segment $\Delta_i \cap \Delta_j$, if non-empty, appears at most once, namely in $Q(\Delta_{max\{i,j\}})$. Similarly, $\sum_{i=1}^{\lceil n/2 \rceil} t_i \le t$. Summing (1) and the contributions of the boundary faces over all blue triangles Δ_i we obtain

$$O\left(\sum_{i=1}^{\lceil n/2 \rceil} \left[(s_i + c_i)^{2/3}t_i^{1/3} + p_i \log p_i\right]\right) =$$

$$O\left(\sum_{i=1}^{\lceil n/2 \rceil} (s_i + c_i)^{2/3}t_i^{1/3}\right) + O(p \log p),$$

which by Hölder's inequality, is

$$O\left(\left[\sum_{i=1}^{\lceil n/2 \rceil} (s_i + c_i)\right]^{2/3} \times \left[\sum_{i=1}^{\lceil n/2 \rceil} t_i\right]^{1/3}\right) + O(p \log p) =$$

$$O\left(\left[\sum_{i=1}^{\lceil n/2 \rceil} s_i + \sum_{i=1}^{\lceil n/2 \rceil} c_i\right]^{2/3} t^{1/3}\right) + O(p \log p).$$

Since P contains m points, it is impossible to make more than $m - 1$ cuts, each of which splits a cell containing more than one point of P into two subcells each containing at least one point of P. Hence $\sum_i s_i \le m - 1$. As to $\sum_i c_i$, we prove in Appendix A the following fact:

Proposition A.2. In an incremental construction of an arrangement of triangles with p intersecting pairs, as above, the total number of faces that cut cells without splitting them is at most $O(p)$.

Recall that by assumption $m \ge 2p+3n$, so $\sum_i s_i + \sum_i c_i \le m + O(p) = O(m)$. Putting everything together, and repeating the argument for the blue faces, we see that the total increase in the number of faces of the purple cells is at most $O(m^{2/3}t^{1/3} + p \log p)$. We have thus shown:

Lemma 1. (Combination Lemma for Arrangements of Triangles) *Let G be a set of n triangles in $\mathbb{R}^3$, let p and t be the number of pairs and of triples of triangles of G which intersect, and let P be a set of m points, with at least one point of P lying on each exposed segment of the arrangement $A(G)$. Partition G into two sets G_1, G_2, and denote by ρ (respectively, β) the complexity (i.e., the number of faces) of all cells of $A(G_1)$ (respectively, $A(G_2)$) marked by P. Then the total complexity of the cells of $A(G)$ marked by P is at most*

$$\rho + \beta + O(m^{2/3}t^{1/3} + p \log p).$$

The recurrence relation for $C(m,n,p,t)$ that we want to develop next also depends on the following graph-theoretic lemma. For a hypergraph H, let $e(H)$ be the number of edges in H.

Lemma 2. *Given an arbitrary 3-uniform hypergraph H on n vertices, let $\{A, B\}$ be a partition of its vertices with $|A| = \lfloor n/2 \rfloor$ and $|B| = \lceil n/2 \rceil$. Let H_1 (resp. H_2) be the sub-hypergraph of H spanned by A (resp. B). Then for some choice of A and B*

$$e(H_1) + e(H_2) < \frac{1}{4} e(H).$$

Proof. For simplicity, assume for the remainder of the proof that n is even. Let A be a randomly selected set of $n/2$ vertices of H (such that all sets of size $n/2$ are chosen with equal probability). Let B be the set of remaining vertices. For an edge e in H,

$$\Pr[e \text{ in } H_1] = \Pr[e \text{ in } H_2] = \binom{n-3}{n/2-3} \Big/ \binom{n}{n/2} < \frac{1}{8}.$$

In particular, the expected number of edges in H_1 and H_2 together is

$$\exp[e(H_1) + e(H_2)] = e(H) \times (\Pr[e \text{ in } H_2] + \Pr[e \text{ in } H_2]) < \frac{1}{4} e(H),$$

implying the existence of a choice of A and B with $e(H_1) + e(H_2) < e(H)/4$, as asserted. ∎

Using the above result in conjunction with the Combination Lemma, we deduce:

Theorem 3. *If G is a set of n triangles in $\mathbb{R}^3$ with p intersecting pairs and t intersecting triples and P is a set of m points, with at least one point of P lying on each exposed segment of the arrangement $A = A(G)$, the total complexity of all cells of A marked by P is*

$$C(m,n,p,t) = O(m^{2/3}t^{1/3} \log n + n + p \log p \log n).$$

Proof. Observe that the total complexity of an arbitrary arrangement of triangles is proportional to the number of its vertices, which is easily seen to be $O(n + p + t)$. Thus $C(m,n,p,t)$ must satisfy the following recurrence relation:

$$C(m,n,p,t) \le \begin{cases} a(n+p+t) & \text{if } m \ge n+p+t, \\ C(m,n/2,p_1,t_1) + C(m,n/2,p_2,t_2) + & \\ \quad b(m^{2/3}t^{1/3} + p \log p) & \text{otherwise,} \end{cases}$$

where a and b are constants, and p_1, t_1, p_2, t_2 are the numbers of pairs and triples of intersecting triangles within each of the two subcollections G_1, G_2, respectively. Lemma 2 applied to the triple-intersection hypergraph of the triangles (i.e., the hypergraph whose nodes are triangles and which contains an edge $\{\Delta_i, \Delta_j, \Delta_k\}$ whenever $\Delta_i \cap \Delta_j \cap \Delta_k \neq \emptyset$ for distinct i, j, k) implies that there is a partitioning of G into two subsets G_1 and G_2 each of size roughly $n/2$ such that $t_1 + t_2 \leq \frac{1}{4}t$. Also trivially $p_1 + p_2 \leq p$.

We claim that

$$C(m,n,p,t) \leq dm^{2/3}t^{1/3}\log n + en + fp\log p \log n, \tag{2}$$

for some constants d, e, f, depending on a and b. First, note that if $m \geq n+p+t$, (2) is trivially satisfied provided $d, e, f > a$. Let us turn our attention to the general case. For the recurrence to be satisfied, the following inequality must hold:

$$\begin{aligned} dm^{2/3}t^{1/3}\log n + en + fp\log p\log n \geq \\ dm^{2/3}t_1^{1/3}\log(n/2) + en/2 + fp_1\log p_1\log(n/2) + \\ dm^{2/3}t_2^{1/3}\log(n/2) + en/2 + fp_2\log p_2\log(n/2) + \\ b(m^{2/3}t^{1/3} + p\log p). \end{aligned}$$

The terms linear in n cancel and $p_1, p_2 \leq p$, so it is sufficient to ensure that the following two inequalities hold:

$$dt^{1/3}\log n \geq dt_1^{1/3}\log(n/2) + dt_2^{1/3}\log(n/2) + bt^{1/3} \tag{3}$$

$$fp\log p\log n \geq fp_1\log p_1\log(n/2) + fp_2\log p_2\log(n/2) + bp\log p. \tag{4}$$

First of all,

$$\begin{aligned} fp_1\log p_1\log(n/2) + fp_2\log p_2\log(n/2) + bp\log p \\ \leq fp\log p\log(n/2) + bp\log p \\ \leq fp\log p\log n + (b-f)p\log p. \end{aligned}$$

Thus (4) holds, provided that $f > b$. Turning to (3), an application of Hölder's inequality yields

$$t_1^{1/3} + t_2^{1/3} \leq 2^{2/3}(t_1+t_2)^{1/3} \leq 2^{2/3}\left(\frac{t}{4}\right)^{1/3} = t^{1/3}.$$

In particular,

$$\begin{aligned} dt_1^{1/3}\log(n/2) + dt_2^{1/3}\log(n/2) + bt^{1/3} &\leq dt^{1/3}\log(n/2) + bt^{1/3} \\ &\leq dt^{1/3}\log n + (b-d)t^{1/3}, \end{aligned}$$

ensuring that (3) holds for any $d > b$, and thus completing the proof. ∎

For developing a lower bound, it will be convenient to have the following graph-theoretical result:

Lemma 4. *Let H be an undirected graph with E edges and C 3-cycles. Then $C = O(E^{3/2})$.*

Proof. Immediate from the following theorem that gives the maximum number C_r of complete subgraphs on r vertices contained in a graph with E edges (in our case, $r = 3$).

Theorem (Erdős [16]).

$$\text{Let } E = \binom{s}{2} + q, \quad 0 < q \leq s, \text{ then } C_r \leq \binom{s}{r} + \binom{q}{r-1}. \qquad \blacksquare$$

Having proved Theorem 3, we obtain our main results concerning all interesting cells by substituting $m = 3n + 2p$ and then observing that $p = O(n^2)$ and $t = O(n^3)$:

Theorem 5.

(a) *The complexity of all interesting cells in an arrangement of n triangles with p intersecting pairs and t intersecting triples is*

$$O((n+p)^{2/3}t^{1/3}\log n + n + p\log p \log n),$$

which reduces to $O(p^{2/3}t^{1/3}\log n)$, if $t \geq p\log^3 n$ and $p \geq n$. Moreover, for any $t \geq p \geq n$ such that $t = O(p^{3/2})$ there exists an arrangement of n triangles with $O(p)$ intersecting pairs and $O(t)$ intersecting triples so that the complexity of all interesting cells is

$$\Omega(p^{2/3}t^{1/3} + p\alpha(p) + n).$$

(b) *The said complexity is also*

$$O((n+p)^{2/3}p^{1/2}\log n + p\log p\log n + n),$$

regardless of the value of t; this is bounded by $O(p^{7/6}\log p)$, provided $p \geq n$.

(c) *In the worst case, $\Phi(n) = O(n^{7/3}\log n)$. This is almost tight, since $\Phi(n) = \Omega(n^{7/3})$.*

Proof. The upper bounds in (a) and (c) are immediate from the preceding analysis, while the bound in (b) easily follows from that in (a) by substituting $t = O(p^{3/2})$, which is a consequence of Lemma 4 applied to the intersection graph of the triangles (note that a triple intersection of triangles is necessarily a 3-cycle in this graph, while the converse is not true).

Turning to the lower bound in (a), let us assume $t \geq p \geq n$ and put $a = \lfloor t/p \rfloor$ and $b = \lfloor p^2/t \rfloor$. Notice that $p^2 \geq t$ by assumption, so both a and b are positive integers. To obtain the lower bound in (a), consider a set of a vertical rectangles whose bottom edges lie in the plane $z = 0$, whose top edges lie in the plane $z = b + 1$, and such that their xy projections yield a planar arrangement of a segments for which the complexity of all interesting (i.e., non-convex) faces is $\Omega(a^{4/3})$[15]. Cutting these rectangles by b additional horizontal (and sufficiently large) rectangles lying in the planes $z = i$, for $i = 1, \ldots, b$, is easily seen to yield an arrangement of $n = a + b$

rectangles (that can be transformed into an arrangement of n triangles in general position) in which the total complexity of all interesting cells is

$$\Omega(a^{4/3}b) = \Omega\left(\frac{t^{4/3}}{p^{4/3}} \times \frac{p^2}{t}\right) = \Omega(p^{2/3}t^{1/3}).$$

Notice that the number of pairwise intersections in this arrangement is no more than $a^2 + ab = t^2/p^2 + p = O(p)$ (as $t^2 = O(p^3)$ by assumption), while the number of triples of intersecting triangles is at most $a^2b \leq t$. This completes the construction of an arrangement of triangles with $O(p)$ intersecting pairs and at most t intersecting triples in which the complexity of all interesting cells is $\Omega(p^{2/3}t^{1/3})$. In addition, note that the interesting cells always have complexity $\Omega(n)$. Moreover, the construction in [23] of an arrangement of n triangles in which the unbounded cell has complexity $\Omega(n^2\alpha(n))$ can easily be modified to yield a similar construction with the unbounded cell having complexity $\Omega(p\alpha(p))$. Putting the three bounds together, we obtain the lower bound in (a). The lower bound in (c) follows by substituting $t = \Theta(n^3)$ and $p = \Theta(n^2)$. ∎

Remark. The reader should note the strong dependency of our bounds on the bounds for planar arrangement of segments. Any future improvements in the two-dimensional bounds (in the lower-order term) will most likely carry over to our analysis and yield corresponding improvements in our bounds.

3. The Slicing Theorem

In this section we will prove an auxiliary result on triangulating cells in arrangements of triangles. It facilitates a general efficient algorithm for computing a single cell in such an arrangement (cf. Section 4). A similar technique is also used to establish better bounds and construct more efficient algorithms for some special classes of arrangements discussed below. In addition, our result provides a partial answer to the following seemingly simple question: given an arbitrary polyhedral region K in $\mathbb{R}^3$, can it be cut into a small number of (pairwise disjoint) tetrahedra? The two-dimensional analog of this problem has a satisfactory solution — the number of triangles needed to triangulate an arbitrary planar polygonal region is proportional to its complexity. In three dimensions, the problem becomes much more difficult. Some negative results are given in [22, Chap. 10]; our result shows roughly that the required number of tetrahedra is not much larger than the complexity of K, if K is a cell (or a collection of cells) in an arrangement of triangles. The technique that we use in the proof of the Slicing Theorem is related to that recently employed by Chazelle and Palios [4] in obtaining similar results for general polyhedra.

Theorem 6. (Slicing Theorem) *Let K be a collection of cells in an arrangement of n triangles in 3-space, with a total of h faces. Then K can be decomposed ("triangulated") into $O(n^2\alpha(n) + h)$ tetrahedra having pairwise disjoint interiors.*

Proof. We construct the triangulation incrementally, by adding new vertical faces (to which we refer as *walls*) emanating from exposed edges of the given triangles. These vertical walls will collectively decompose the cells of K into convex *subcells*. Once this is done, the final triangulation is easily obtained by triangulating the boundary of each convex subcell C and connecting each resulting triangle to some fixed interior point of C. The number of final tetrahedra is clearly proportional to the total complexity of all subcells into which the walls partition K, which we now proceed to estimate.

First of all, we assume that *all cells of K are interesting*; the above observations imply that this involves no real loss of generality. Let $e_1, \ldots, e_{3n}$ be the exposed edges of the triangles. For each e_i in turn we add vertical walls emanating from it, as follows. Suppose this has already been done for all $e_j, j < i$. Let p_i be the vertical plane containing e_i, and let V_i be the collection of segments in p_i, each of which is either the intersection of p_i with some triangle or a connected component of the intersection of p_i with some previously erected wall. Then the walls added at the current (ith) stage are simply all the *horizon faces* of e_i (i.e., faces incident to e_i) in the planar arrangement $A(V_i)$ which are contained in K. See Fig. 4 for an illustration. (For convenience of exposition, we will need to assume that no exposed edge is vertical and that $p_i \neq p_j$ whenever $i \neq j$; in absence of collinear exposed edges, this can always be enforced by an appropriate rotation — collinear edges can be handled by a slight modification of this argument.)

It is easy to check inductively that the addition of all these walls results in a convex decomposition of K. Indeed, the walls added at the ith stage remove all non-convex edges occurring on e_i, and no new non-convexities are created.

We next analyze the total complexity (i.e. the number of faces) of the resulting decomposition of K. It is easily seen that this complexity is $h + O(q)$, where q is the total complexity (i.e., total number of edges) of all the vertical walls. A major obstacle in estimating q is that we have no *a priori* linear bound on the number of segments in each V_i — each triangle contributes at most one segment to V_i, but a vertical wall intersects p_i in many segments and there are many vertical walls erected from each e_j. We overcome this difficulty as follows:

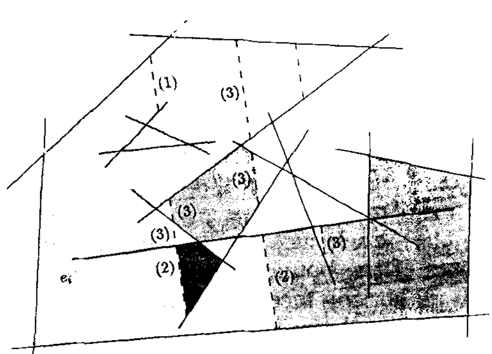

Fig. 4. The arrangement $A(V_i)$ in the vertical plane p_i containing exposed edge e_i. Solid segments represent intersections of p_i with triangles, while three previously erected sets of walls intersecting p_i are dashed. The shaded regions represent new walls. Vertical segments are marked with their type

Fix an exposed edge e_i, and let V_i^0 be the collection of the (at most n) segments formed by intersecting p_i with the given triangles. By the result of [10], the total complexity of all horizon faces of e_i in $A(V_i^0)$ is $O(n\alpha(n))$. Let H be the collection of all horizon faces of e_i in $A(V_i)$ lying within K, namely the vertical walls erected from e_i. Clearly, each face in H is a subface of some horizon face in $A(V_i^0)$. Note that, for each $j < i$, the vertical walls erected from e_j intersect p_i in a collection of segments lying in the vertical line $p_i \cap p_j$ (see Fig. 4) so that, in particular, at most one of them cuts e_i. It follows that the horizon faces in $A(V_i)$ meet e_i in at most $4n-1$ edges, because the segments of V_i cut e_i in at most $4n-2$ points (there are at most $3n-1$ planes p_j with $j < i$ and at most $n-1$ triangles not containing e_i). Since each such edge bounds at most two faces, we conclude that there are at most $8n-2$ horizon faces in $A(V_i)$. We will subdivide each segment of $V_i - V_i^0$ into subsegments, cutting it at its point of intersection with segments of V_i^0, and work with the refined vertical segments from now on. For simplicity assume that all segments are finite; however, the argument that follows can be easily adapted to handle rays and full lines. Since each vertex of a face in H is either a vertex of a horizon face of e_i in $A(V_i^0)$ (of which there are only $O(n\alpha(n))$), or an endpoint of some vertical segment in $V_i - V_i^0$, it therefore suffices to estimate the total number of such segments which bound horizon faces of e_i in $A(V_i)$. We now add these vertical segments one at a time, thus gradually transforming the horizon faces in $A(V_i^0)$ into those in $A(V_i)$. Let V_i^j denote the current collection of segments obtained after adding j of these segments. Each segment s in $V_i - V_i^j$ bounding a final horizon face in $A(V_i)$ is of one of the following types (note that, by construction, each endpoint of a finite vertical segment lies on a segment of V_i^0):

(1) s bounds the same horizon face in $A(V_i^j \cup \{s\})$ on both of its sides,

(2) s lies on the common boundary of two distinct horizon faces of $A(V_i^j \cup \{s\})$, or

(3) s bounds, on one side, a horizon face of $A(V_i^j \cup \{s\})$ and on the other side a face of $A(V_i^j \cup \{s\})$ which is no longer adjacent to e_i.

Refer to Fig. 4. Observe that, as a new vertical segment is added to V_i^j to obtain V_i^{j+1}, the type of the remaining segments may change; in fact, it is easily verified that the only transitions possible are (1)→(2), (2)→(3), and (1)→(3). We proceed to transform the horizon faces of $A(V_i^0)$ into those of $A(V_i)$ by first adding all segments of $V_i - V_i^0$ which at the time of addition have type (1). (As noted, no transition can revert a segment to type (1).) Each such segment reduces the number of "islands" (i.e., connected components of the boundary) of some horizon face by 1. But the total number of islands in all horizon faces is at most n, because each segment of V_i appears in at most one island, and (on the assumption that no vertical segment is infinite) each island must contain a segment of V_i^0. Thus the total number of type (1) segments added is at most n. A similar argument shows that the number of type (2) segments, which we add after the type (1) segments, is also $O(n)$, because addition of each such segment increases the number of horizon faces by 1, and there are only $O(n)$ such faces in the final $A(V_i)$. The remaining vertical segments all have type (3) and will never change their type again. Let V_i^* be the union of V_i^0 with all type (1) and type (2) vertical segments. Since $|V_i^*| = O(n)$, it is still the case that the complexity of all horizon faces of e_i in $A(V_i^*)$ is $O(n\alpha(n))$. Now add the vertical segments of type (3). Each such segment s, in its turn, chops off some portion f_s of a horizon face, which becomes disconnected from e_i. Moreover, since we add only segments that actually appear on the boundaries of the final horizon faces in $A(V_i)$ and vertical segments are pairwise disjoint, each portion f_s is chopped off only once (namely, these portions have pairwise disjoint interiors) and, since f_s is a polygonal region, it must contain a vertex of a horizon face in $A(V_i^*)$ other than either of the endpoints of s. Hence the number of portions f_s, and thus also the number of type (3) segments s, is $O(n\alpha(n))$. Repeating the analysis for each e_i, we conclude that the total number of faces in the decomposition of K if $h + O(n^2\alpha(n))$; as asserted. As discussed above, a final triangulation of this decomposition completes the proof of the theorem. ∎

Corollary 7. *Let K be a collection of q cells having a total h faces, in an arrangement of n triangles in $\mathbb{R}^3$. Then K can be decomposed into $q+O(n^2)$ convex polyhedra with pairwise disjoint interiors, such that their total complexity is only $h + O(n^2\alpha(n))$.*

Proof. It is sufficient to demonstrate that in the above construction the number of resulting subcells does not exceed q by more than $O(n^2)$, since the second claim follows immediately from the proof of the Slicing Theorem. Recall that a vertical wall w erected at a given step of the construction is a face in the arrangement induced in its plane by intersections with all triangles and all vertical walls erected in previous steps. Hence w cuts the subcell C in which it is contained into at most two new subcells (note that it is possible for w to change the topology of the boundary of C without splitting C into two subcells). Hence introduction of w increases the number of subcells by at most 1. However, there are only $O(n)$ vertical walls erected from each exposed edge, thus the total number of convex polyhedra obtained by the construction does not exceed q by more than $O(n^2)$. ∎

Corollary 8. *The collection K of all interesting cells in an arrangement of n triangles in 3-space can be decomposed into $O(n^2)$ convex polyhedra with pairwise disjoint interiors, such that their total complexity is only $h + O(n^2\alpha(n))$, where h is the complexity of K.*

Proof. Immediate from Corollary 7, since there are at most as many interesting cells as there are exposed segments, i.e. $O(n^2)$. ∎

Remarks. (1) While the Slicing Theorem is intuitively plausible, the proof is not at all trivial. An open problem is to generalize the theorem to the case of an arbitrary 3-dimensional semi-algebraic set K, namely to show that K can be decomposed into a collection of simple, connected cells (e.g., such as in Collins' cylindrical algebraic decomposition [7]), whose cardinality is roughly the same as the combinatorial complexity of ∂K. A partial affirmative result of this sort is indicated in [6] for the collection K of *all* 3-cells in an arrangement of spheres; the analysis given there appears to be extendible to arbitrary algebraic surfaces, provided, however, that we still consider all cells of the arrangement. A related open problem is to obtain generalizations of the Slicing Theorem to arrangements of simplices in higher dimensions.
(2) Observe that the key point of the proof of the Slicing Theorem is the assertion that, given a planar arrangement A of n "old" segments and an unspecified number of "new" segments, the total complexity of the faces of A cut by a segment e is still $O(n\alpha(n))$, provided the new segments (i) cross e at most $O(n)$ times, (ii) do not intersect among themselves, and (iii) start and end at points of old segments or "at

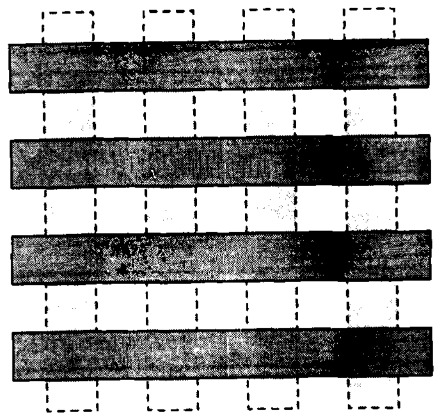

Fig. 5. A collection of disjoint rectangles in which our "slicing" produces $O(n^2)$ polyhedra. Solid rectangles lie in the plane $z = 1$, dashed ones in $z = 0$

infinity". It is easy to verify that the proof given above applies to this slightly more general situation as well.
(3) Note that even when the number p of pairs of intersecting triangles in the arrangement is $o(n^2)$, the number of cells in the resulting decomposition of K can still be quadratic (this is the case, for example, in the arrangement of n long and thin horizontal rectangles, $n/2$ of which lie in the plane $z = 0$, while the remaining $n/2$ lie in the plane $z = 1$, as shown in Fig. 5; here in fact $p = 0$).

4. Special Cases

In this section we will discuss two restricted classes of arrangements of triangles for which the worst-case complexity of a single cell (or of all interesting cells) is easier to bound.

4.1. Triangles with Few Orientations

We first consider arrangements of n triangles in which the planes containing the triangles have at most f distinct orientations. (Note that in the following analysis f need not be constant.) A (very restricted) example of such an arrangement would be when each triangle lies in a plane parallel to one of the $f = 3$ coordinate planes. In what follows, we will assume however, without loss of generality, that no triangle lies in a plane parallel to the z axis. Corollary 8 yields a decomposition of all interesting cells in such an arrangement into $O(n^2)$ polyhedra. Being convex, each of these polyhedra is bounded by at most two faces contained in planes having any specific orientation, so it is bounded by at most $2f$ faces lying in the original triangles. The remaining faces lie in the vertical walls added in the construction of the Slicing Theorem; thus they do not contribute to the complexity of the (undecomposed) interesting cells (except by cutting other faces into subfaces). Hence, the total number of non-vertical faces in all polyhedra lying in interesting cells is $O(fn^2)$, which is an upper bound on the total number of faces in all interesting cells. This proves

Theorem 9. *The complexity of all interesting cells in an arrangement of n triangles lying in planes with f distinct orientation is at most $O(fn^2)$.*

As for a corresponding lower bound, it is easy to obtain a variant of the construction of [15] for a lower bound on the complexity of many faces in arrangements of lines, which involves f^3 segments in the plane, having only f distinct orientations, such that the complexity of all interesting (i.e., non-convex) faces of their arrangement is $\Omega(f^4)$. Assume $f \le (n/2)^{1/3}$. Placing $\lfloor n/2f^3 \rfloor$ disjoint translates of such an arrangement side by side in the plane produces an arrangement of at most $n/2$ segments with f distinct orientations, in which the complexity of all interesting faces is $n/2f^3 \times \Omega(f^4) = \Omega(fn)$. Now consider a 3-dimensional arrangement of at most $n/2$ vertical rectangles whose lower edges lie in the plane $z = 0$, whose upper edges lie in the plane $z = \lfloor n/2 \rfloor + 1$, and whose vertical projections induce the planar arrangement of at most $n/2$ segments described above, together with $\lfloor n/2 \rfloor$ large horizontal rectangles lying in the planes $z = i$, for $i = 1, \dots, \lfloor n/2 \rfloor$. It is easy to see that the total complexity of the interesting cells in this arrangement of n rectangles is at least $n \times \Omega(fn) = \Omega(fn^2)$, but only $f+1$ distinct plane orientations were used. Moreover, the rectangles can be easily replaced by triangles without reducing the combinatorial complexity of the interesting cells. Thus we have shown

Theorem 10. *The worst-case complexity of all interesting cells in an arrangement of n triangles lying in planes with f distinct orientations is at least $\Omega(fn^2)$ for $f = O(n^{1/3})$.*

A similar argument, based on the construction of a planar arrangement of n segments in which the unbounded face has complexity $\Omega(n\alpha(n))$ [28], gives

Theorem 11. *The worst-case complexity of a single cell in an arrangement of n triangles lying in planes with f distinct orientations is at least $\Omega(n^2\alpha(f))$.*

Corollary 12. *In an arrangement of n triangles lying in planes with a constant number of distinct orientations, the worst-case complexity of any single cell (or of all interesting cells) is $\Theta(n^2)$.*

4.2. The Horizontal-Vertical Case

The second special case concerns arrangements where only two types of convex plates are allowed: arbitrary (convex) horizontal polygons (i.e., parallel to the xy-plane) and vertical rectangles (i.e., rectangles with two sides parallel to the z axis). Let n be the total number of plates (both horizontal and vertical) in the arrangement and let q be the total complexity of all horizontal plates. The lower bound construction of Theorem 5 applies in this case, showing that the total complexity of all interesting cells in this restricted class of arrangements is still $\Omega(n^{7/3})$. We now use a variant of the decomposition given by the Slicing Theorem to prove a tight

$\Theta(n^2\alpha(n) + q)$ bound on the worst-case complexity of a single cell in such an arrangement. Recall that the only type of non-convexity that occurs in an interesting cell is along an exposed segment. We distinguish two types of interesting cells:

(1) those whose boundary contains only vertical exposed segments (i.e., those parallel to the z axis) and

(2) those whose boundary contains horizontal exposed segments (and possibly some vertical ones).

We will argue that the complexity of a single cell of type (1) is $O(n\alpha(n))$, while the *total* complexity of type (2) cells is $O(n^2\alpha(n)+q)$, concluding that the complexity of any one cell in such an arrangement is $O(n^2\alpha(n) + q)$.

Consider a cell C of type (1). Clearly, it cannot be unbounded. Moreover, the absence of horizontal exposed segments on ∂C implies that C is in fact a right vertical prism whose height is a maximal vertical open segment extending from a "floor plate" Δ_f to a "ceiling plate" Δ_c and whose bases are identical faces in the planar arrangements $Q(\Delta_f)$ and $Q(\Delta_c)$ induced in Δ_f and Δ_c, respectively, by their intersections with the vertical rectangles. $Q(\Delta_f)$ and $Q(\Delta_c)$ each contain at most n segments, so the complexity of a single face in either of the two arrangements is $O(n\alpha(n))$. Since C is a prism, its complexity is proportional to the complexity of either of its bases, implying our first claim.

There are two types of horizontal exposed segments on the boundaries of type (2) cells: portions of the horizontal edges of vertical rectangles and portions of edges of the horizontal plates. We will restrict our attention to the latter; the former are handled similarly. Let p be the plane containing some horizontal plate Δ. Intersections of vertical rectangles with p induce a planar arrangement $Q = Q(\Delta)$ in p of at most n segments. Refine the type (2) cells by adding a *wall* filling each face of Q that lies outside of Δ and is incident to the relative boundary $\partial\Delta$ of Δ. Repeating the same procedure for every horizontal plate (and every horizontal edge of a vertical rectangle, within the horizontal plane containing it), we obtain a decomposition of all cells of type (2) into subcells. Note that all horizontal exposed segments have been eliminated, so each subcell is of type (1) (and thus is a right vertical prism). We will now argue that the bases of subcells contained in type (2) cells have total complexity $O(n^2\alpha(n))$, so that the total complexity of these subcells is bounded by $O(n^2\alpha(n))$ as well; this would complete the proof of the upper bound as the only elements of type (2) cells not accounted for in the combined complexity of their subcells is the number of horizontal exposed segments and their incident faces and vertices, and there are $O(n^2 + q)$ exposed segments altogether, with each exposed segment incident to exactly two faces and two vertices.

Consider a subcell C' of a type (2) cell. Let f and c be the floor and ceiling bases of C' contained in horizontal planes p_f and p_c, respectively. By construction, $\partial C' - f \cup c$ must consist of portions of the vertical rectangles extending all the way from f to c. In particular, any cross section of C' by a horizontal plane p is a face in the planar arrangement $Q(p)$ of segments induced in p by its intersections with the vertical rectangles. Therefore f (respectively, c) is the union of one or more such faces in an analogous arrangement $Q(p_f)$ in p_f (respectively, $Q(p_c)$ in p_c), which is identical to $Q(p)$ except for the addition of the edges of the plate contained in p_f (resp. p_c). Observe that, by construction, the closure of either f or c (or both) must contain a horizontal exposed segment, for otherwise C' would coincide with an original cell of type (1). Let us suppose that it is f that meets some horizontal exposed edge. Thus f coincides with the union of some horizon faces of the boundary $\partial\Delta_f$ of the convex plate Δ_f in the arrangement $Q(p_f)$. Moreover, it is easily checked that each such horizon face in $Q(p_f)$ lies in the base of at most two subcells (in the floor of one cell and in the ceiling of another). Now the results of [10] are easily seen to imply that the horizon of any closed convex curve in a planar arrangement of n segments has complexity $O(n\alpha(n))$; thus, the combined complexity of the bases of all subcells of type (2) cells is $O(n^2\alpha(n))$, as asserted.

The lower bound of $\Omega(n^2\alpha(n) + q)$ for the complexity of a single cell is easily obtained by a slight modification of the construction of Theorem 5. Thus we have shown

Theorem 13. *In an arrangement of n horizontal convex plates and vertical rectangles, the worst-case complexity of a single cell is $\Theta(n^2\alpha(n)+q)$, where q is the total number of corners of the horizontal plates.*

5. Algorithms

Here we come to the issue of actually building castles in the air, as promised in the title of this paper. Namely, we are interested in computing one or more cells in an arrangement A of n triangles in $\mathbb{R}^3$. Note that it is not difficult to compute the *full arrangement* A in time $O(n^3)$ by modifying the algorithm of [14] for constructing the arrangement of n *planes*. Thus our goal is to attain subcubic performance in computing a single cell (or all interesting cells). The previous results of [23] are non-algorithmic, and our combinatorial analysis cannot be immediately translated into an efficient algorithm either. In section 5.1 we describe a randomized algorithm for computing a single cell in an arbitrary arrangement of n triangles; its expected time complexity is asymptotically nearly optimal in the sense described below. Moreover, the same algorithm takes only $O(n^{2+\delta})$ time, for any $\delta > 0$ (with the constant of proportionality depending on δ), when applied to arrangements of vertical rectangles and horizontal convex plates discussed in the previous section (assuming that each horizontal plate has constant complexity), or to arrangements of n triangles lying in planes with a constant number of distinct orientations. We then comment on the difficulty of extending our algorithm to compute many cells, and outline an alternative, less efficient but still subcubic algorithm for obtaining all interesting cells, which is based on the recent results of [9]. Finally, we present a customized deterministic algorithm for arrangements of triangles with few distinct plane orientations (as in Section 4).

All of our algorithms use a common data structure to represent cells in three-dimensional arrangements of triangles. A cell is represented by the list of its boundary components, starting with the explicitly marked outer boundary (which may be empty) and followed by zero or more "islands", all linked to one another (the exact nature of this linkage will be discussed below). Each boundary component is stored in the form of an "adjacency structure", where each face is given by its outer boundary (unless the face is unbounded) plus zero or more islands, properly linked to one another, each of which is in turn represented as a circular list of incident edges and vertices. Conceptually, we store each face twice (corresponding to the two sides of the triangle), leading to similar multiplicities in the representation of edges and vertices, as in the puffy model discussed in Section 2.1. This ensures that every boundary element is incident to a *unique* cell (duplicate elements are still linked among themselves; in fact an actual implementation need not store them as physically distinct entities). Thus each (copy of an) edge is incident to exactly two faces, a triple-intersection vertex is incident to three faces and three edges, etc. The precise implementation of such a data structure can be done along the lines suggested in [8] or in [27]; however we will not be concerned here with such implementation details.

5.1. Calculating a Single Cell

Our algorithm proceeds as follows:

Input: A set G of n triangles in $\mathbb{R}^3$, a point p, and an "efficiency parameter" $\delta > 0$.

Output: The cell of $A(G)$ containing p, represented by the incidence structure discussed above.

(1) Select a random sample R of r triangles of G (where r is a constant, depending on δ, to be determined later).

(2) Construct the full arrangement $A(R)$ (using any convenient, even brute-force, method).

(3) By exhaustive search, identify the cell C_R of $A(R)$ that contains p.

(4) Subdivide C_R into open tetrahedra as in the Slicing Theorem (again, using brute force).

(5) Let t_0 be the tetrahedron containing p. For each triangle $\Delta \in G$ compute the convex polygon $\Delta \cap t_0$, and let G_{t_0} be the collection of such (triangulated) non-empty polygons. Apply the algorithm recursively to (G_{t_0}, p, δ) to obtain a cell C_0. If C_0 is a bounded cell of $A(G_{t_0})$, stop — C_0 is the desired cell of $A(G)$ containing p. Otherwise apply the trimming step (8) below to $C_{t_0} = C_0$ (denoting the resulting trimmed subcell containing p by $\hat{C}_0$).

Repeat steps (6)–(8) for each of the remaining tetrahedra $t \neq t_0$ in the decomposition of C_R and proceed to step (9).

(6) As for t_0, compute for each triangle $\Delta \in G$ the convex polygon $\Delta \cap t$, and let G_t be the collection of all such non-empty (triangulated) polygons.

(7) Apply the algorithm recursively to (G_t, p_∞, δ) where p_∞ is "the point at infinity", to obtain the unbounded cell C_t of $A(G_t)$.

(8) Trim C_t along the faces of t and discard the outer portion of the cell; this may split C_t into several subcells all lying within t and incident to the boundary ∂t of t.

(9) Reconstruct the cell C of $A(G)$ containing p by starting with $\hat{C}_0$ and "growing" it as follows: Let a *window* of a (trimmed) cell C' of $A(G_t)$ be a face of C' on ∂t introduced by the trimming step (8). Clearly each window is shared by two cells lying in adjacent tetrahedra. A window is *transparent* if it is not contained in a triangle of G. Starting with $\hat{C}_0$, locate all cells sharing transparent windows

with it and "glue" them to $\hat{C}_0$ along the windows. Repeat until the resulting cell C has no unpaired transparent windows. This is the desired cell.

Why does the algorithm work? Observe that the cell C_G of $A(G)$ containing p is completely contained in the cell C_R of $A(R)$ containing p, so that the algorithm correctly restricts its computation to C_R. Notice that if p does not lie in the unbounded cell C_{t_0} of $A(G_{t_0})$ then C_G must coincide with C_0, in which case no further computation is necessary and the algorithm correctly returns C_0. Assume $p \in C_{t_0} = C_0$ which is unbounded. If a point x lies in $C_G - \hat{C}_0$, there is a path σ in C_G connecting it to p. A traversal of σ from p to x starts off in $\hat{C}_0$ and proceeds to visit trimmed cells of $A(G_t)$ (for various t), moving from cell to cell through transparent windows. Moreover each cell C crossed by σ must be a portion (within the corresponding tetrahedron t) of the unbounded cell C_t of $A(G_t)$, because C can be reached from outside t. Hence x will be in the cell built by the algorithm. So the constructed cell is no smaller than C_G and, trivially, it cannot be larger than C_G either. Thus the cell the algorithm computes is precisely C_G.

We next analyze the expected running time of the algorithm. Since r is a constant, all steps of the algorithm besides recursive invocations and the trimming and reconstruction steps can be performed in overall linear time. Denoting by $|C|$ the complexity of a cell C, observe that trimming the unbounded cell C_t of $A(G_t)$, for some tetrahedron t, can be performed in time proportional to $|C_t|$, as it is sufficient to check each feature of C_t against each face of t. The time required for the reconstruction step is proportional to the total complexity of all transparent windows, because this step can be accomplished by simultaneously tracing the common face of each pair of adjacent tetrahedra using the incidence structure of the trimmed cells. Determining whether C_0 is the unbounded cell of $A(G_{t_0})$ can be accomplished in constant time, since the representation we use explicitly stores the outer component (if any) of a cell boundary. Hence the total time complexity of all steps of the algorithm, excluding recursive calls, is bounded by $An + B\sum_t |C_t|$, for some positive constants A, B depending on r, with the summation taken over all tetrahedra t contained in C_R. Observe that a small perturbation of $\cup_t G_t$ (that moves tetrahedra away from each other) produces an arrangement of $\sum_t |G_t|$ triangles in which the boundary of the unbounded cell is precisely $\cup_t \partial C_t$. In particular, $\sum_t |C_t| \leq \zeta(\sum_t |G_t|)$. It immediately follows that the time complexity of the algorithm exclusive of recursive calls is bounded above by $O(n + \zeta(\sum_t |G_t|))$. Now $\sum_t |G_t|$ is easily seen to be bounded by Dn, for some constant D depending on r. Therefore, the algorithm requires $O(\zeta(Dn))$ overhead and recurs on K problems $(G_{t_i}, p_\infty, \delta)$, where K is the total number of tetrahedra contained in C_R (by the Slicing Theorem, $K \leq J\zeta(r)$ where J is a constant *independent* of r), and t_i varies over all such tetrahedra (including t_0). By the ϵ-net theory of [19] (see also [5]), with high probability, each tetrahedron meets at most $\frac{an}{r}\log r$ triangles, for some constant $a > 0$ independent of r. Moreover, with no extra overhead we can verify that this does indeed hold for our sample R; if not we simply discard R and try another sample, until we hit one which has the ϵ-net property. It easily follows that the expected number of such iterations is a constant (depending on r). Hence, in expected linear time, the algorithm produces a sample for which $|G_{t_i}| \leq \frac{an}{r}\log r$, for all i. In particular, denoting by $T(n)$ the expected running time of the algorithm, we obtain the following recurrence for $T(n)$:

$$T(n) \leq \begin{cases} KT(\frac{an}{r}\log r) + O(\zeta(Dn)), & \text{if } n > r, \\ O(r^3) & \text{otherwise.} \end{cases}$$

Define $\gamma = \limsup_{h\to\infty} \log\zeta(h)/\log h$. Recall that $\zeta(n)$ has been shown to be $\Omega(n^2\alpha(n))$ and $O(n^{7/3}\log n)$, so $2 \leq \gamma \leq 7/3$. Hence, for any $\delta > 0$, there is a choice of a constant b such that the term $O(\zeta(Dn))$ is bounded above by $bn^{\gamma+\delta/2}$ for all n (with b depending on δ and r). Thus the recurrence reduces to

$$T(n) \leq \begin{cases} KT(\frac{an}{r}\log r) + bn^{\gamma+\delta/2}, & \text{if } n > r, \\ O(r^3) & \text{otherwise,} \end{cases}$$

which has a solution $T(n) \leq Hn^\epsilon$ for any $\epsilon > \max\{\gamma + \delta/2, \log K/\log(r/a\log r)\}$ (with H depending on r, δ and ϵ). Notice that the second quantity can be made not to exceed $\gamma + \delta/2$ by choosing a sufficiently large r, as $K \leq J\zeta(r)$. Choosing such an r for the given input value of δ, we obtain $T(n) = O(n^{\gamma+\delta})$, with the constant depending on δ. In particular, we have shown:

Theorem 14. *Given a fixed $\delta > 0$, a single cell in an arrangement of n triangles in $\mathbb{R}^3$ can be computed in randomized expected time $C_\delta n^{\gamma+\delta}$, with the constant C_δ depending on δ, where $\gamma = \limsup_{k\to\infty}\log\zeta(k)/\log k \in [2, 7/3]$.*

Corollary 15. *A single cell in an arrangement of n triangles can be computed in randomized expected time $O(n^{7/3+\delta})$, for any $\delta > 0$, with the constant of proportionality depending on δ.*

Remark. Note that the expectation in the time bound of our algorithm is over the random selection of the sample R, and *not* over any distribution of the input. Our algorithms thus have the same asymptotic expected complexity for any general arrangement of n triangles.

Notice that, should the conjecture posed in the introduction be demonstrated to hold, the above theorem would immediately provide an $O(n^{2+\delta})$ expected-time algorithm for computing any single cell of the arrangement.

We also note that the polygons passed to recursive invocations of the algorithm will not in general be triangles. However, it is easy to see that, independently of the level of recursion on which such a polygon is created, it can be represented as the intersection of an original triangle with a single tetrahedron. Hence such a polygon can have no more than 7 sides, and the presence of non-triangles can be compensated for by using $\zeta(5n)$ instead of $\zeta(n)$ in our argument (as any convex polygon with up to 7 sides can be cut into at most 5 triangles), and by replacing the constant a by $5a$.

Remark. Several recent techniques (due to Clarkson [5], Chazelle and Friedman [3], Matoušek [20]; and Aggarwal) provide, in certain special cases, tools either for a deterministic construction of a triangulation of space with properties similar to those discussed above, or for obtaining such a decomposition in which each cell is cut by only $O(n/r)$ objects, rather than $O((n\log r)/r)$. We do not know whether these techniques can be adapted to the problem at hand, so as to make the above algorithm deterministic or further improve its running time.

Finally, observe that the time complexity analysis of the above algorithm relies on the following facts:

1. The complexity of any cell in arrangement of n triangles is bounded by a function $\zeta(n)$ such that a cell in the arrangement formed by any subset of r triangles can be decomposed into $O(\zeta(r))$ tetrahedra.
2. With high probability, the number of triangles cutting any tetrahedron that is missed by the r sample triangles is at most $(an \log r)/r$, for some constant a independent of r and n.
3. The intersection of a tetrahedron with a triangle is a convex polygon of constant complexity.

Thus, in a situation where the above conditions hold (perhaps with a different function ψ taking the place of ζ), the algorithm will correctly compute the desired cells in expected time $O(\psi(n)n^{\delta})$, for any $\delta > 0$. In particular, we have:

Theorem 16. *Given a fixed $\delta > 0$, a single cell in an arrangement of n triangles lying in planes with a constant number f of distinct orientations can be computed in randomized expected time $C_{\delta,f}n^{2+\delta}$, with the constant $C_{\delta,f}$ depending on δ and f.*

Theorem 17. *Given a fixed $\delta > 0$, a single cell in an arrangement of n horizontal triangles and vertical rectangles can be computed in randomized expected time $C_{\delta}n^{2+\delta}$, with the constant C_{δ} depending on δ.*

Observe that Theorem 16 is superceded by Theorem 19 of Section 5.3 which, for a constant f, provides a faster and more general *deterministic* algorithm for computing portions of such arrangements.

5.2. Calculating Many Cells

An attempt to extend the algorithm of Section 5.1 to calculate an arbitrary collection of cells of an arrangement of triangles (e.g. all interesting cells) faces the following technical problem: our algorithm takes advantage of the fact that the desired single cell is contained in a single cell of the arrangement $A(R)$ of the random sample of triangles. This implies that the number of tetrahedra that require further processing is only $O(r^{7/3} \log r)$ (in fact, $O(\zeta(r))$), which leads to a recurrence with a favorable time complexity. In contrast, when calculating many cells, there is no *a priori* sharp bound on the overall complexity of the cells of $A(R)$ which contain the desired cells of $A(G)$ — in the worst case all cells of $A(R)$ could be involved — so that no comparable recurrence relation can be obtained. Hence, although our algorithm can be easily adapted to calculate a collection of cells, we do not have sharp worst-case bounds on its expected running time in this case.

Instead, we use the following alternative approach. Edelsbrunner et al. [9] recently described a randomized procedure which, given a planar arrangement Q of n segments, preprocesses it in expected time

$$O(n^{5/3} \log^{5/2} n)$$

and space $O(n^{4/3}\alpha(n) \log n)$ so that, given any query point x, one can calculate the face of Q containing x in time $O(n^{1/3} \log^{11/2} n + k)$, where k is the complexity of that face. Using this procedure, a subset of the cells of $A(G)$ can be computed as follows.

Consider first the task of calculating all interesting cells. We begin by obtaining a point on each exposed segment (in overall $O(n^2)$ time). Consider each of these points y in turn, and let Δ be a triangle containing y. The algorithm proceeds to trace ∂C, where C is the cell containing y, as follows. It first obtains the face f containing y in the arrangement induced in Δ by intersections with the remaining triangles, using the procedure of [9], and then moves to adjacent faces of ∂C, which share an edge with f and lie in other triangles, obtains each of them, and repeats this procedure until the entire boundary component is traced this way. One then needs to "hop" from one boundary component of C to another. This is done by "linking" triangles in the following fashion: In the preprocessing stage, for each of the $3n$ triangle corners, one computes (by brute force) the triangle that lies directly below it (if any) and the point of intersection of this triangle with a vertical line through the corner. Given such a point z lying directly below a corner c, one proceeds (again, by brute force) to the boundary of the face containing z (in the induced planar arrangement) and then follows the boundary until the first vertex c' of that face is encountered; then c' is linked to c and vice versa. Corners that have no triangle below them are linked to the "vertex at infinity". The desired "boundary-hopping" is then accomplished by examining all vertices in the computed boundary component and following the links from any of them to vertices that have not yet been encountered, which then serve as starting points y in new boundary components. Consider an interior boundary component K of a bounded cell. Since the lowest vertex of K is a triangle corner, it is easy to verify that *some* link connects K to either the outer boundary component or to an interior component whose lowest point is below that of K. This immediately implies that all interior components are (possibly indirectly) linked to the outer component. This, in particular, allows one to pass from one boundary component of a cell to another until all have been traced. The argument in the case of the unbounded cell is similar, with the "vertex at infinity" playing the role of the outer component of the boundary. It is easy to verify that linear time is sufficient to compute each of the links by brute force, thus only quadratic additional time suffices to guarantee that no component of the desired boundary will be left unvisited.

Consider next the general case where we want to calculate an arbitrary collection of cells, each specified by a point x in its interior. For each such cell C, we first identify some point y on its boundary ∂C. This can be done in linear time per cell by choosing y to be the first point of intersection of the downward-directed vertical ray emanating from x with any triangle; if there is no such point, let y be the "vertex at infinity" — C in this case is the unbounded cell and the algorithm will then start at the vertex at infinity and follow all precomputed links from it to boundary components of C (as in the discussion just presented).

If the number m of points marking cells is very large, this linear overhead per marker x may be too expensive. In this case we can locate the desired points y as follows. Extend each triangle into a full plane, and apply the (randomized) algorithm of [11] which calculates the plane lying immediately below each of these m points, in expected time $O(m^{3/4-\delta}n^{3/4+3\delta}\log^2 n)$ for any $\delta > 0$. We thus obtain m points y, each lying below a corresponding given point x on one of the n planes containing the triangles. Using the precomputed data structures of [9] for fast queries, we compute, for each y, a vertex on the boundary of the face f containing y in the arrangement induced in the plane of the corresponding triangle, thereby obtaining a vertex on the boundary of the cell containing y (and, therefore, x). This requires $O(n^{1/3}\log^{11/2} n)$ expected time per query.

Having obtained in this manner an "anchor vertex" on the boundary of each of the desired cells, we can then continue as in the case of all interesting cells described above. We omit further details concerning this extension. Thus we obtain

Theorem 18. *Any collection of cells of $A(G)$, defined by specifying a set of m points marking the desired cells, can be computed in expected time*

$$O(m^{3/4-\delta}n^{3/4+3\delta}\log^2 n + (M+m)n^{1/3}\log^{11/2} n + n^{8/3}\log^{5/2} n)$$

and $O(m^{3/4-\delta}n^{3/4+3\delta} + M + n^{7/3}\alpha(n)\log n)$ working storage, for any $\delta > 0$ (with the constant of proportionality depending on δ), where M is the total complexity of the desired cells. If the number m of marking points is small, a more straightforward approach will compute the marked cells in expected time

$$O(mn + Mn^{1/3}\log^{11/2} n + n^{8/3}\log^{5/2} n)$$

using $O(M + n^{7/3}\alpha(n)\log n)$ space. A somewhat simpler algorithm computes all interesting cells of $A(G)$ in expected time

$$O(n^{8/3}\log^{14/3} n)$$

and space $O(n^{7/3}\alpha(n)\log n)$.

Remark. Notice that the amount of work required for computing an arbitrary subset of cells is largely dominated by the preprocessing time, which in turn requires nearly as much time as that for computing all interesting cells. Therefore, it may be assumed that all interesting cells are to be calculated anyway and one is facing the problem of computing some additional dull (i.e. convex) cells of an arrangement of triangles, where each cell is specified by a point in it.

5.3. A Deterministic Algorithm for Arrangements with Few Orientations

In this section we sketch an efficient $O((m+M+n^2)f\log n)$-time deterministic algorithm for computing m cells in an arrangement of n convex plates lying in planes with only f distinct orientations, where M is the total complexity of the cells being computed. For simplicity of presentation we will assume that the plates are in fact triangles and that no more than three triangles meet at a common point, but a more general situation can be handled just as easily. Each desired cell C is identified by a point $x \in C$. Given x, the algorithm locates a vertex v_0 on ∂C, using a technique similar to that in Section 5.2, but admitting a much simpler implementation. Specifically, the plane defined by a triangle and lying immediately below x can be determined in $O(f\log n)$ time by performing f binary searches, one in each collection of parallel planes. Once the point y lying in such a plane directly below x is located, a point on ∂C is easily obtained by one $O(f\log n)$-time ray-shooting query (described below) in that plane, and a vertex on ∂C can be obtained after two more queries of this kind. Then it traces out all edges and vertices of ∂C

by starting at v_0 and repeatedly "shooting" along an edge of ∂C incident to the "current" vertex v and thus discovering a new vertex w of ∂C. This gives us two new edges incident to w and we shoot along them to discover further vertices.

Two specific problems arise in this approach. First, how can each "shot" be performed in $O(f \log n)$ time? Shooting along exposed segments is easily implemented by precomputing the answers to all possible queries, while each ray-shooting query along a line of intersection of two plates reduces to a two-dimensional ray-shooting in some arrangement induced in either plate by its intersections with the remaining plates. By assumption, each such arrangement A_i in plate Δ_i is formed by at most $f-1$ overlaid subarrangements, each of which consists of a collection of parallel segments. Moreover, there are only $f-1$ possible directions for the shooting ray. Thus, for each subarrangement and each shooting direction we can prepare a data structure that supports $O(\log n)$ shooting queries using, for example, the technique of [25]. It is easily checked that this approach guarantees $O(f \log n)$ query time while the preprocessing can be accomplished for all n plates in $O(fn^2 \log n)$ time and $O(fn^2)$ storage. (This also takes care of the initial step of moving from each given point x to a vertex on the boundary of the cell containing it.)

The second problem that arises is handling clusters of edges lying in ∂C but not connected to each other. Such situations can occur if there are "islands" in faces of ∂C, or if ∂C is not connected. We have dealt with the latter case in Section 5.2 by linking each triangle corner to (a vertex lying in) the triangle directly below it to facilitate "hopping" from one component of ∂C to another. The former situation is approached similarly except that the linking is performed in each planar arrangement A_i induced in the plate Δ_i by intersections with the remaining plates. For each segment endpoint c in A_i, we locate the segment s lying immediately below c (in Δ_i). Then c is linked to the vertex of A_i lying on s nearest to the point of s immediately below c. If the "linking" steps are carried out using ray-shooting, the total overhead can be shown not to exceed $O(fn^2 \log n)$.

Theorem 19. *Given a set of m marking points, the cells marked by these points in an arrangement of n triangles lying in planes with f distinct orientations can be computed in deterministic time $O((m+M+n^2)f \log n)$ and $O(M+fn^2)$ storage, where M is the total complexity of the marked cells. In particular, if no two markers lie in the same cell, the time complexity of the algorithm reduces to $O((M+n^2)f \log n)$.*

6. Discussion and Open Problems

The paper leaves one major open problem, namely the settling of our conjecture that $\zeta(n) = \Theta(n^2\alpha(n))$. In addition, we note the following extension and applications of our results:

(1) As already mentioned, the analysis of Section 2 easily extends to yield a similar upper bound for the complexity of all non-convex cells in an arrangement of n arbitrary flat convex plates in 3-space (with an additive correction term that accounts for the complexity of the plate boundaries). We omit the straightforward details of this extension.

(2) We believe that our results can be generalized to higher dimensions; specifically we conjecture that the maximum number of facets (highest-dimensional faces)

of all non-convex cells in an arrangement of n d-simplices in $(d+1)$-dimensional space is close to $O(n^{d+1/3})$. (Note that the lower bound construction can be generalized to any number of dimensions, yielding a lower bound of $\Omega(n^{d+1/3})$ for this complexity.) The technique used in [11] to obtain such an extension for the case of hyperplanes may be useful for proving this conjecture.

(3) The technique we used to prove the Combination Lemma can be adapted to yield simpler proofs for other combination lemmas, such as those given in [11,12]. Appendix B exemplifies this claim by providing a simple proof for a variant of the two-dimensional combination lemma of [12] for faces in arrangements of segments.

(4) Expanding upon our motion-planning application, we note that our analysis shows that the accessible portion C of the configuration space FP of a general, not necessarily convex, polyhedron B with K faces translating amidst polyhedral obstacles having n faces altogether, has

$$O((Kn)^{7/3}\log(Kn))$$

combinatorial complexity, and that it can be calculated in expected time

$$O((Kn)^{7/3+\delta})$$

(more precisely, in time $O(\zeta(Kn)(Kn)^{\delta}))$, for any $\delta > 0$. However, to actually plan a motion between two given placements z_1, z_2 of B, it may not even be necessary to calculate the entire component C of FP containing these placements. In this regard we claim that if B can translate from z_1 to z_2, then it can always do so along a polygonal path having only $O((Kn)^2)$ turns, and that in the worst case (at least for a constant K) that many turns are necessary. The claim follows immediately from the fact that C can be cut into $O((Kn)^2)$ convex polyhedra, as shown in Section 3 (Corollary 7 to the Slicing Theorem). An example where $\Omega(Kn^2)$ turns are necessary (for some non-convex B) is not difficult to construct. However, we do not know how to calculate such a path efficiently, without obtaining first the entire component C. Further extensions and applications of the combinatorial results obtained above to motion planning problems are discussed in a forthcoming paper [2].

(5) An interesting by-product of the Slicing Theorem and the preceding remark is that the techniques of [26] for coordinated motion planning for two independent robots can be applied to obtain an $O(n^{14/3}\log^2 n)$ (actually, $O(\zeta(n)^2)$) algorithm for coordinating the motions of two independently translating polyhedra, each having $O(1)$ complexity, amidst stationary polyhedral obstacles with a total of n faces. See [26] for more details.

(6) Let us conclude by considering again the Conjecture posed in the introduction. Theorem 3 suggests a generalized and weakened version of the conjecture which states that the complexity of any m cells of $A(G)$ is

$$O(m^{2/3}t^{1/3}\log n + n + p\log p\log n).$$

where p (resp. t) is the number of pairs (resp. triples) of intersecting triangles in G. In particular, putting $m = 1, p = O(n^2)$, and $t = O(n^3)$ in this conjectured bound would yield $\zeta(n) = O(n^2\log^2 n)$. Does this bound on $\zeta(n)$ (slightly weaker than originally conjectured) hold?

Appendices

A. Topology of Arrangements of Triangles

In this appendix we derive several basic properties of the topological structure of an arrangement of n triangles in $\mathbf{R}^3$. We begin with the analysis of the number of cutting-but-not-splitting faces, required in the proof of the Combination Lemma and proceed to justify our practice of bounding the complexity of a cell by the number of its faces. Our arguments make use of the "puffy" model of the arrangement, mentioned in Section 2, and further discussed below.

The Genus of Cell Boundaries

In this section we will analyze the topological structure of the union of a family of triangles in $\mathbf{R}^3$ in order to bound the number of "cutting-but-not-splitting" faces in the proof of the Combination Lemma.

For the following analysis, it will be useful to regard the boundary of a cell in an arrangement of triangles in $\mathbf{R}^3$ as a compact orientable 2-manifold. To this end, we replace each triangle Δ in the arrangement by a "puffy triangle" Δ^* which is a thin nearly flat body bounded by two surfaces, one on each side of Δ, slightly deformed away from one another. Δ^* has the same edges and vertices as Δ, and is sufficiently thin so that the combinatorial pattern of intersections of the puffy triangles is identical to that of the original arrangement (such choice of thicknesses is always possible under the general position assumption). Cells in the new arrangement are connected components of the common exterior (i.e., complement of the union) of all puffy bodies. However, it is easy to check that now cell boundaries are compact orientable 2-manifolds (without boundary). Note that in this puffy model, every original face occurs in the new arrangement exactly twice, each (pairwise intersection) edge — four times, each (triple-intersection) vertex — eight times, and each vertex that is the intersection of an exposed edge with another triangle — twice. Triangle corners and exposed segments are not duplicated.

Let us recall some facts from elementary algebraic topology (see [17] for reference). For a cell complex X, let $\chi(X)$ be its Euler characteristic. It has the property that, for any two cell complexes X and Y

$$\chi(X \cup Y) = \chi(X) + \chi(Y) - \chi(X \cap Y) \tag{5}$$

(if $X \cap Y$ is a sub-complex of X and Y). We will require the following standard facts: $\chi(S^1) = 0$, $\chi(S^1 \times I) = 0$, $\chi(S^2) = 2$, $\chi(D^2) = 1$, where $I = [0, 1]$ is the unit interval, S^1 — a circle, S^2 — a sphere, D^2 — a closed disk, and $S^1 \times I$ — a cylinder (more precisely, the surface of a bounded cylinder without the "lids"). In particular, notice that Euler characteristic is additive when X and Y are disjoint or intersect along a set of disjoint circles and/or cylinders. Also recall that the Euler characteristic of a compact connected orientable 2-manifold (without boundary) is equal to $2(1 - g)$, where g is its genus.

We now return to arrangements of triangles. Consider a triangle Δ cutting an arrangement $A = A(G)$ of n other triangles. The intersection of Δ with A is a planar arrangement $Q = Q(\Delta)$ of (at most) n segments all contained within Δ.

Keeping in mind the "puffy" model of the arrangement, observe that each face f of Q corresponds to two faces in the arrangement $A' = A(G \cup \{\Delta\})$, and that the introduction of this pair of faces changes some cell C of A (either by splitting it into two subcells, or by just modifying its boundary ∂C). Our goal is to measure the effect of adding f on the genus of ∂C. For the remainder of this discussion, we will assume that Δ *does* intersect some triangle of G, for otherwise it simply adds an extra component (of genus 0) to ∂C without changing the rest of ∂C. Let us denote by g (respectively, g') the genus summed over all connected components of ∂C (respectively, $\partial C'$, where C' is the one or two cells produced by cutting C with f), and by c (respectively, c') the number of such connected components. As we will only be interested in the differences $\delta c = c' - c$ and $\delta g = g' - g$, we may assume that the quantities g, c, g', and c' refer only to the components of ∂C and $\partial C'$ met by f. Also notice that, modulo the above assumption, the Euler characteristic of ∂C can be computed as $\chi(\partial C) = 2(c - g)$ (simply by summing χ over all components of ∂C separately and observing that the Euler characteristic is additive here as the components are disjoint by definition). A similar identity holds for $\chi(\partial C')$. Two cases are possible: either

(i) f touches the relative boundary $\partial\Delta$ of Δ, in which case the introduction of f cannot split C into two subcells, or

(ii) f lies within the relative interior of Δ, in which case C may or may not be split by f into two disjoint cells.

Recall that the boundary of f consists of a *single* outer component and zero or more *islands*, all enclosed by the outer boundary and enclosing mutually disjoint regions (that lie outside of f).

We will consider case (i) first. Let the outer boundary of f be denoted $\partial_{out} f$. Suppose that f has $i \geq 0$ islands and that $\partial_{out} f - \partial\Delta$ consists of $b \geq 0$ connected components. Since we have assumed that Δ intersects some triangles of G, at least one of i, b is not zero. In the puffy model, f can be identified with a sphere with $2i + b$ open discs deleted (two discs are deleted for each island — one on either side of f — and one disk for every connected component of $\partial_{out} f - \partial\Delta$). Using (5), we obtain

$$\chi(f) = \chi(S^2) - (2i + b)\chi(D^2) + (2i + b)\chi(S^1) = 2 - (2i + b).$$

Now observe that the boundary $\partial C'$ of the new cell C' can be obtained by removing b disks (one for each contact of $\partial_{out} f$ with ∂C) and i cylinders (one for each island of f) from ∂C and "gluing" the remaining portion to f along the $2i + b$ "seam" circles. The Euler characteristic of ∂C with b disks and i cylinders removed is easily seen to be

$$\chi(\partial C) - i\chi(S^1 \times I) - b\chi(D^2) + (2i + b)\chi(S^1) = \chi(\partial C) - b.$$

Applying (5) once again, we obtain the Euler characteristic of $\partial C'$:

$$\chi(\partial C') = (\chi(\partial C) - b) + \chi(f) - (2i + b)\chi(S^1) = \chi(\partial C) + 2(1 - i - b).$$

Recalling the relation between the genus and the Euler characteristic of a compact orientable 2-manifold, we conclude $2(c' - g') = 2(c - g) + 2(1 - i - b)$, or $\delta g = \delta c + i + b - 1$. Since c' (the number of components of $\partial C'$ met by f) is 1 and $1 \leq c \leq i + b$, it follows that $1 - i - b \leq \delta c \leq 0$ and thus $0 \leq \delta g \leq i + b - 1$.

Let us now consider the case (ii) where f is an internal face of Δ with i islands, so that f has two sides that are not directly connected to each other. More precisely, each side of f is a topological disk with i open disks removed, so

$$\chi(f) = 2[\chi(D^2) - i\chi(D^2)] = 2(1-i).$$

Once again, to obtain the boundary $\partial C'$ of the new cell(s) we have to delete $i+1$ cylinders from ∂C (one for each component of the boundary of f) and glue it to f along the $2(i+1)$ "seam" circles. the Euler characteristic of ∂C with $i+1$ cylinders removed is $\chi(\partial C)$. "Gluing" it to f along the $2(i+1)$ circles, we obtain

$$\chi(\partial C') = \chi(\partial C) + \chi(f) = \chi(\partial C) + 2(1-i).$$

Recalling the expression for the Euler characteristic of the components of ∂C and $\partial C'$ met by f, we obtain $2(c'-g') = 2(c-g) + 2(1-i)$, so that $\delta g = \delta c + i - 1$. Here c' is either 2 or 1 depending on whether f splits C into two cells or not. (Clearly, if C is split in two, $c' = 2$. Otherwise, consider the connected component K of $\partial C'$ containing one side of f. If it includes the opposite side of f, $c' = 1$, as asserted. If it does not, observe that K is a compact connected oriented 2-manifold (without boundary); thus its removal disconnects $\mathbb{R}^3$ into an interior and an exterior components (see, for example, [21, Chap. 26]). As K does not include the other side of f, two points lying sufficiently close to each other on opposite sides of f are separated by $K \subset \partial C''$, making it impossible for both of them to lie in the same cell, and thus contradicting the assumption that C was not split in two.) On the other hand, c can be as high as $i+1$ or as low as 1 (f has one outer boundary and i islands; the extreme cases correspond to all of them lying in different components of ∂C or all lying in the same component). Hence $-i \le \delta c \le 1$ and $-1 \le \delta g \le i$. Moreover, if f does not split C into two cells, $c' = 1$ and the above relation becomes $\delta g = i - c$. In particular, we have shown that each cutting-but-not-splitting internal face f must be of one of the following two types:

(1) $i = 0$ and $\delta g = -1$, or

(2) $i > 0$.

To summarize the cases: introduction of a face f (in $Q(\Delta)$) either reduces the total genus by one or increases it by (at most) the number of its islands (if f is a boundary face of $Q(\Delta)$, it may also cause an additional increase in the total genus by as much as the number of components of $\partial_{out} f - \partial\Delta$ less 1). However, the number of islands over all faces of $Q(\Delta)$ is easily seen to be bounded by the total number p_Δ of segments in $Q(\Delta)$ (since a segment cannot appear in two islands) and the number of components of $\partial_{out} f - \partial\Delta$ over all faces f in $Q(\Delta)$ is bounded by $2p_\Delta$. Hence:

Theorem A.1. *The change in the total genus of all cell boundaries in an arrangement A of triangles, caused by adding a triangle Δ to A, is between $3p_\Delta$ and $-k$, where k is the number of internal faces of $Q(\Delta)$, having no islands, whose introduction does not split a cell of A into two subcells, and p_Δ is the number of segments in $Q(\Delta)$.*

Thus, summing these changes in an incremental construction of $A(G)$, as in proof of the Combination Lemma in Section 2, we obtain:

Proposition A.2. *In an incremental construction of an arrangement of triangles, as carried out in the proof of the Combination Lemma, the total number of faces that* *cut cells without splitting them is at most $O(p)$, where p is the number of pairs of triangles that have non-empty intersection.*

Proof. We will count the number of such faces in a construction that starts with no triangles and proceed to add first all red and then all blue triangles, one at a time. This clearly provides an overestimate on the desired quantity. The total genus of the boundaries of all cells is always non-negative and the genus of an empty set (the boundary of $\mathbb{R}^3$ — the only cell in an arrangement of no triangles) is zero. It is decreased by one by the introduction of each type (1) internal face. On the other hand, the previous theorem implies that the overall increase in the total genus is proportional to the total number of segments in all arrangements $Q(\Delta)$, which is to say, $O(p)$. Hence the total number of faces for which $\delta g = -1$ is $O(p)$. As islands cannot be shared among faces of $Q(\Delta)$, there are at most p_Δ faces of type (2) in $Q(\Delta)$, for a grand total of $O(p)$. The conclusion follows. ∎

Theorem A.3. *The genus summed over the boundaries of all cells in an arrangement of triangles in which p pairs intersect is at most $O(p)$.*

Proof. Immediate from Theorem A.1. ∎

Euler's Relationship for Cell Boundaries

In this section we will justify our practice of bounding the complexity of a cell by the number of its faces. Consider a single connected component K of a cell boundary. By using the puffy model, we ensure that K is a compact connected orientable 2-manifold, and thus we can apply standard topological techniques to relate the number of faces, edges, and vertices in K to its Euler characteristic and genus (see, for example, [17]). Since faces of K are not necessarily simply connected, we must account for this as well. Let v_K, e_K, f_K be the number of vertices, edges, and faces of K, respectively. If all faces of K were simply connected, the definition of the Euler characteristic $\chi(K)$ of K would yield

$$v_K - e_K + f_K = \chi(K).$$

Consider a multiply connected face — its boundary consists of a single outer component and zero or more islands. Notice that introduction of i cuts transforms an i-island face into a simply-connected one, where a cut is a simple arc connecting a vertex of an island to a vertex of the outer boundary of the face. Thus all non-simply-connected faces of K can be eliminated by increasing the number of edges in K by the total number i_K of "islands" on the faces of K. Hence

$$(6) \qquad v_K - (e_K + i_K) + f_K = \chi(K) \qquad \text{or} \qquad v_K + f_K = \chi(K) + e_K + i_K.$$

Notice that all vertices of K besides triangle corners have degree three or more, while triangle corners have degree two. Denoting the number of such corners on K by c_K and summing the degree over all vertices, we obtain

$$(7) \qquad 2c_K + 3(v_K - c_K) \le 2e_K \qquad \text{or} \qquad 3v_K \le 2e_K + c_K.$$

Conditions (6) and (7) together imply

$$e_K \le c_K + 3f_K - 3\chi(K) \qquad \text{and} \qquad v_K \le c_K + 2f_K - 2\chi(K).$$

Recall that the Euler characteristic of a compact connected orientable manifold is $2(1 - g_K)$, where g_K is the genus of K, yielding

$$e_K < c_K + 3f_K + 6g_K,$$

and a similar bound for v_K. Summing these inequalities over all connected components K of ∂C (with C being a fixed cell or any collection of cells), we obtain

$$e_{\partial C} \leq c_{\partial C} + 3f_{\partial C} + 6g_{\partial C} = O(f_{\partial C} + n + p)$$

and

$$v_{\partial C} \leq c_{\partial C} + 2f_{\partial C} + 4g_{\partial C} = O(f_{\partial C} + n + p),$$

where $g_{\partial C}$ and $c_{\partial C}$ refer to the total genus of all components of ∂C (in the sense of the last section) and the total number of triangle corners on these components, respectively. The last equality follows from Theorem A.3 and the observation that there are $3n$ triangle corners in the whole arrangement. Thus we have shown:

Theorem A.4. *The total combinatorial complexity of a collection of cells in an arrangement of n triangles in space is $O(F + p + n)$, where F is the total number of faces in the boundaries of these cells and p is the number of intersecting pairs of triangles.*

B. Other Combination Lemmas — An Example

In this appendix we demonstrate the strength of the technique that we used in proving the Combination Lemma (Lemma 1) by employing it to obtain a simple proof of a variant of the combination lemma of [12] for faces in arrangements of segments in $\mathbb{R}^2$.

Lemma B.1. *Consider two arrangements of segments, red and blue, with a total of n segments. Assume that no two segments intersect in more than a point. Let $\mathcal{B} = \{B_1, \ldots, B_s\}$ (resp. $\mathcal{R} = \{R_1, \ldots, R_t\}$) be a family of faces in the red (resp. blue) arrangement. Let P be a set of k points $\{p_1, \ldots, p_k\}$, such that each point p_i lies in $B_{s_i} \cap R_{t_i}$ for some unique s_i and t_i. For each $i = 1, \ldots, k$, let E_i denote the face of the combined arrangement (i.e., the connected component of $B_{s_i} \cap R_{t_i}$) containing p_i. Then the complexity of the "purple" family $\{E_i\}$ is at most $\beta + \rho + O(k + n)$, where β, ρ are the complexities of $\mathcal{B}, \mathcal{R}$, respectively, and the complexity of a face containing two or more points of P is still counted only once.*

Proof. Let n_r and n_b be the number of red and blue segments, respectively. We follow the general approach of the proof of Lemma 1, but the analysis here is considerably simpler. That is, starting with the red arrangement, we will incrementally superimpose the blue segments on it, thereby trimming and subdividing the red faces in $\mathcal{R}$ until they assume their final "purple" shape. In the process we will bound the number of additional *red* edges that are created on purple boundaries. Repeating the same process, starting with the blue family $\mathcal{B}$ and incrementally transforming it to the purple family, will provide an estimate on the number of additional blue edges bounding the E_i's. Together, these bounds give an upper bound on how much the complexity of $\{E_i\}$ exceeds $\rho + \beta$.

We thus begin with the red family $\mathcal{R}$ and add blue segments, one by one. At any step during this process, we refer to the polygonal regions of the current planar map that contain points of P as "currently purple". Consider the next blue segment e to be added to the arrangement. We will refer to a maximal segment of the intersection of e with a currently purple face as a *blue fragment*. Clearly the next currently purple family may be obtained by adding all blue fragments of e to the currently purple regions and then considering the regions of the resulting planar map which contain points of P. Consider adding blue fragments to the current arrangement one at a time. Let b be a blue fragment (contained in some currently purple face Q) being added in the current step. By construction, the relative interior of b does not meet any currently purple face boundaries. Therefore, b can be classified into exactly one of the following five classes:

(i) b cuts Q into two subfaces, each containing points of P;

(ii) b cuts off a portion of Q containing no point of P;

(iii) b connects two components of ∂Q, without fully cutting Q;

(iv) b meets ∂Q at only one of its endpoints;

(v) b is a full blue segment lying entirely in the interior of Q.

(Note that the class of a fragment is determined *at the time of insertion.*) We now proceed to estimate the total number of blue fragments in each class over the course of construction and the amount of additional red complexity that they create. In case (v), introduction of b clearly has no effect on the red complexity of currently purple faces, so let us restrict our attention to blue fragments of types (i)-(iv). Each fragment of type (i) or (iii) creates at most two new red edges (as it meets ∂Q exactly twice). There are at most $k-1$ type (i) fragments, as each such fragment further refines the partition of P into sets of points lying in distinct currently purple cells. As each type (iii) fragment connects two "islands" of a currently purple cell and no segment (either blue or red) can lie in two islands, type (iii) fragments occur at most $n-1$ times. In particular, there are at most $n_r - 1$ type (iii) fragments, connecting two points of red segments and thus increasing the red complexity by 2, and at most n_b fragments of type (iii) linking a point on a previously inserted blue segment to one on a red segment, each increasing the red complexity by 1. The fragments of type (iii) that connect two points of previously inserted blue segments do not affect the red complexity. As for type (ii) fragments, note that the endpoints of b lie on *different* edges of ∂Q, as b is a line segment and no two segments overlap. Each such edge is cut by the endpoint of b into two subedges, one of which lies outside of the newly truncated currently purple cell. Thus no additional red edges are created. Turning to type (iv) fragments, each of them creates at most two new red edges out of the edge that it meets, thereby increasing the red complexity by no more than one. However, the endpoint of b that does not lie on ∂Q must be, by construction, an endpoint of a blue segment, so that there are at most $2n_b$ type (iv) blue fragments. Thus, once all blue segments have been added, the overall increase in the number of red edges in all purple regions is at most

$$2(k-1) + 2(n_r - 1) + n_b + 2n_b = 2k + 2n_r + 3n_b - 4.$$

Repeating the analysis for the increase in blue complexity, we conclude that the total complexity of all purple faces is at most

$$\beta + \rho + 4k + 5n - 8,$$

as asserted. ∎

References

[1] B. Aronov, H. Edelsbrunner, L. J. Guibas, and M. Sharir: *Improved bounds on the number of edges of many faces in arrangements of line segments.* Manuscript, 1989.

[2] B. Aronov, and M. Sharir: *The union of convex polyhedra and translational motion planning,* in preparation.

[3] B. Chazelle, and J. Friedman: A deterministic view of random sampling and its use in geometry, In *Proc. 29th IEEE Symp. on Foundations of Computer Science,* 1988, 539-549.

[4] B. Chazelle, and L. Palios: *Triangulating a non-convex polytope,* Proc. 5^{th} Symp. on Computational Geometry, **1989**, 393–400.

[5] K. Clarkson: New applications of random sampling in computational geometry, *Discrete Comput. Geom.,* **2** (1987) 195-222.

[6] K. Clarkson, H. Edelsbrunner, L. J. Guibas, M. Sharir, and E. Welzl: Combinatorial complexity bounds for arrangements of curves and surfaces, *Discrete Comput. Geom.,* **5**(1990) 99–160.

[7] G. E. Collins: Quantifier elimination for real closed fields by cylindrical algebraic decomposition, In *2nd GI Conference on Automata Theory and Formal Languages,* Springer, **1975**, 134-183.

[8] D. P. Dobkin, and M. J. Laszlo: Primitives for the manipulation of three-dimensional subdivisions, In *Proc. 3rd ACM Symp. on Computatioal Geometry,* **1987**, 86-99.

[9] H. Edelsbrunner, L. J. Guibas, J. Hershberger, R. Seidel, M. Sharir, J. Snoeyink, and E. Welzl: Implicity representing arrangements of lines or of segments, *Discrete Comput. Geom.,* **4**(1989) 433–466.

[10] H. Edelsbrunner, L. J. Guibas, J. Pach, R. Pollack, M. Sharir, and R. Seidel: Arrangements of curves in the plane: Topology, combinatorics and algorithms, In *Proc. 15th Int. Colloq. on Automata, Languages and Programming,* **1988**, 214-229.

[11] H. Edelsbrunner, L. J. Guibas, and M. Sharir: The complexity of many cells in arrangements of planes and related problems, *Discrete Comput. Geom.,* **5**(1990) 197–216.

[12] H. Edelsbrunner, L. J. Guibas, and M. Sharir: The complexity and construction of many faces in arrangements of lines and of segments, *Discrete Comput. Geom.,* **5**(1990) 161–196.

[13] H. Edelsbrunner, L. J. Guibas, and M. Sharir: The Upper Envelope of Piecewise Linear Functions: Algorithms and Applications, Tech. Report 333, Comp. Sci. Dept., Courant Institute, November **1987**. *Discrete Comput. Geom.* **4**(1989) 311–336.

[14] H. Edelsbrunner, J. O'Rourke, and R. Seidel: Construsting arrangements of lines and hyperplanes with applications, *SIAM J. Computing,* **15** (1986) 341-363.

[15] H. EDELSBRUNNER, and E. WELZL: On the maximal number of edges of many faces in an arrangement, *J. Comb. Theory,* Ser. A. **41** (1986) 159-166.

[16] P. ERDŐS: On the number of complete subgraphs contained in certain graphs, *Publ. Math. Inst. Hungar. Acad. Sci.,* **7** (1962) 459-464.

[17] V. GUILLEMIN, and A. POLLACK: *Differential Topology,* Prentice-Hall, **1974**.

[18] S. HART, and M. SHARIR: Nonlinearity of Davenport-Schinzel sequences and of generalized path-compression schemes, *Combinatorica,* **6** (1986) 151-177.

[19] D. HAUSSLER, and E. WELZL: Epsilon-nets and simplex range queries, *Discrete Comput. Geom.,* **2** (1987) 127-151.

[20] J. MATOUŠEK: Construction of ε-nets, In *Proc* 5^{th} *Symp. on Computational Geometry,* **1989**, 1–10.

[21] E. E. MOISE: *Geometric Topology in Dimensions 2 and 3,* Springer, **1977**.

[22] J. O'ROURKE: *Art Gallery Theorems and Algorithms,* Oxford University Press, **1987**.

[23] J. PACH, and M. SHARIR: The Upper Envelope of Piecewise Linear Functions and the Boundary of a Region Enclosed by Convex Plates: Combinatorial Analysis, *Discrete Comput. Geom.* **4**(1989) 291–310.

[24] R. POLLACK, M. SHARIR, and S. SIFRONY: Separating two simple polygons by a sequence of translations, *Discrete Comput. Geom.,* **3** (1988) 123-136.

[25] N. SARNAK, and R. E. TARJAN: Planar point location using persistent search trees, *CACM,* **29** (7) (1986) 669-679.

[26] M. SHARIR, and S. SIFRONY: Coordinated motion planning for two independent robots, In *Proc. 4th ACM Symp. on Computational Geometry,* **1988**, 319-328.

[27] C. D. THOMBORSON, L. L. DENEEN, and G. M. SHUTE: *A uniform representation for partially embedded graphs,* Manuscript.

[28] A. WIERNIK and M. SHARIR: Planar realization of non-linear Davenport-Schinzel sequences by segments, *Discrete Comput. Geom.,* **3** (1988) 15-47.

Boris Aronov

Courant Institute of
Mathematical Sciences,
New York University, U. S. A.
Current address
Department of Computer Science
Polytechnic University
Brooklyn, NY 11201, U. S. A.

Micha Sharir

Courant Institute of
Mathematical Sciences,
New York University, U. S. A.
and
School of Mathematical Sciences
Tel Aviv University, Israel