Part I

Adversary Lower Bounds

Down with Euclid! Death to the triangles!

— Jean Dieudonné, c. 1960

An adversary means opposition and competition,
but not having an adversary means grief and loneliness.

— Zhuangzi (Chuang-tsu), c. 300 BC

Chapter 2

Affine Degeneracies

A fundamental problem in computational geometry is determining whether a given set of points is in "general position." A simple example of this type of problem is determining, given a set of points in the plane, whether any three of them are colinear. In 1983, van Leeuwen [151] asked for an algorithm to solve this problem in time $o(n^2 \log n)$. Chazelle, Guibas, and Lee [39] and Edelsbrunner, O'Rourke, and Seidel [68] independently discovered an algorithm that runs in time and space $O(n^2)$ by constructing the arrangement of lines dual to the input points.[1] Edelsbrunner *et al.* [68] also solved the higher-dimensional version of this problem, which we call the *affine degeneracy problem* Their algorithm, given n points in $\mathbb{R}^d$, determines whether $d+1$ of them lie on the same hyperplane, in time and space $O(n^d)$.[2] Edelsbrunner and Guibas [65] later improved the space bound to $O(n)$ in all dimensions.

A basic primitive used by all of these algorithms is the *sidedness query*: Given $d+1$ points $p_0, p_1, \ldots, p_d$, does the point p_0 lie "above", on, or "below" the oriented hyperplane $\mathrm{aff}(p_1, \ldots, p_d)$? These are also sometimes called orientation tests, simplex queries, or (in the plane) triangle queries. The result of a sidedness query is given by the sign of

[1] We refer readers unfamiliar with projective duality to [140].

[2] The original analysis of their algorithm was flawed. A correct proof of the crucial Zone Theorem was later given by Edelsbrunner, Seidel, and Sharir [69].

the following determinant.

$$\begin{vmatrix} 1 & p_{01} & p_{02} & \cdots & p_{0d} \\ 1 & p_{11} & p_{12} & \cdots & p_{1d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & p_{d1} & p_{d2} & \cdots & p_{dd} \end{vmatrix}$$

The value of this determinant is $d!$ times the signed volume of the simplex spanned by $p_0, \ldots, p_d$. The *orientation* of a simplex $(p_0, p_1, \ldots, p_d)$ is the result of a sidedness query on its vertices (in the order presented). If the orientation is zero, we say that the simplex is *degenerate*.

In the algebraic decision tree and algebraic computation tree models, there is a somewhat trivial lower bound of $\Omega(n \log n)$ on finding affine degeneracies in any dimension, since it takes $\Omega(n \log n)$ time just to determine whether all the points are distinct [138, 16]. Prior to the results described in this chapter, no better lower bound was known in any model of computation.

Two sets of labeled points are said to have the same *order type* if corresponding simplices have the same orientation. The order type of a set of points can be represented by the face lattice of its dual hyperplane arrangement or by its lambda-matrix [88], both representations requiring space $\Omega(n^d)$. One might consider representing order types by canonical sets of points. Unfortunately, the full field of algebraic numbers is required to represent every planar order type [96], and even among integer order types, point coordinates must be doubly-exponential in the worst case [90].

The fastest known algorithm for determining the order type of a set of points constructs its dual hyperplane arrangement in time and space $O(n^d)$ [68, 69]. Even though all known representations of order type require space $\Omega(n^d)$, there is some hope of a smaller representation, and thus, a faster algorithm, since it is known that there are only $(n/d)^{\Theta(d^2 n)} = 2^{\Theta(n \log n)}$ order types [89]. Prior to the results in this chapter, the information-theoretic lower bound of $\Omega(n \log n)$ was the only lower bound known for this problem.

In this chapter, we first derive a lower bound of $\Omega(n^d)$ on the number of sidedness queries required to decide if a set of n points in $\mathbb{R}^d$ is affinely degenerate, or to determine the set's order type. This matches known upper bounds. Our lower bound holds in a decision tree model of computation in which every decision is based on the result of a sidedness query.

We are not allowed, for example, to compare the values of different sidedness determinants. This is not quite as unreasonable a restriction as it may appear at first glance; all known algorithms for determining degeneracy or order type rely (or can be made to rely) exclusively on sidedness queries [39, 65, 68]. Our lower bound implies that there is no hope of improving these algorithms unless other primitives are used.

These lower bounds follow from an extremely simple adversary argument. We describe a nondegenerate set of points that contains $\Omega(n^d)$ independent "collapsible" simplices, any one of which the adversary can make degenerate without changing the orientation of any other simplex. If an algorithm fails to perform a sidedness query for every collapsible simplex, the adversary can move the points so that the perturbed set is degenerate, and the algorithm will be unable to distinguish between the original set and the perturbed set. The adversary's point set consists of rational points on a particular polynomial curve.

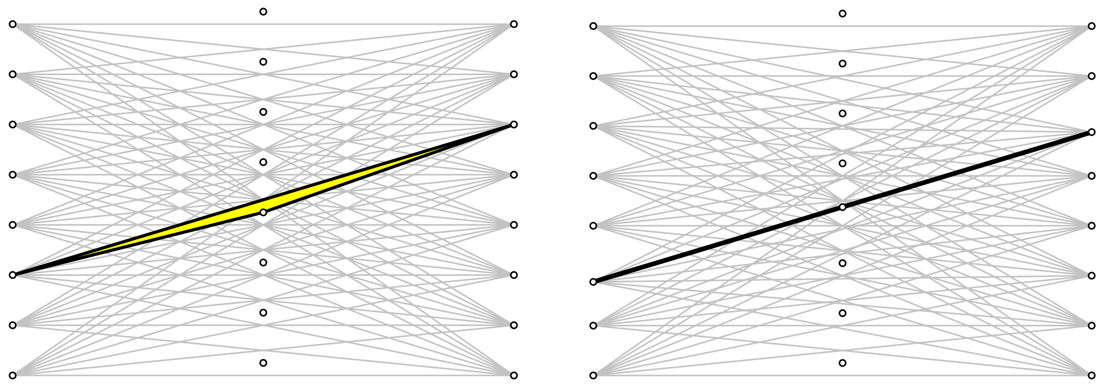
Later in the chapter, we describe a large class of "allowable" primitives, which do not improve the lower bound even by a single sidedness query, even if we permit our algorithms to perform an arbitrarily large (but finite) number of them. Allowable queries include coordinate comparisons, slope comparisons, comparisons of second-order points defined as vertices of the dual hyperplane arrangements, and so forth. In fact, almost every bounded-degree multivariate polynomial is an allowable query.

2.1 Lower Bounds for a Restricted Problem

We begin by considering a restricted version of the degeneracy problem. Say that a hyperplane in $\mathbb{R}^d$ is *vertical* if it contains a line parallel to the x_d axis. The *nonvertical affine degeneracy problem* asks, given a set of n points in $\mathbb{R}^d$, whether there is a nonvertical hyperplane passing through $d+1$ of them. In this section, we prove the following theorem.

Theorem 2.1. *Any decision tree that detects nonvertical affine degeneracies in* $\mathbb{R}^d$, *using only sidedness queries, must have depth* $\Omega(n^d)$.

In order to give a more intuitive picture, we first consider the planar case, and then generalize to arbitrary dimensions.



(a) (b)

Figure 2.1. A planar adversary construction for nonvertical degeneracies. (a) The initial set, with one collapsible triangle shaded. (b) The perturbed set, showing collapsed triangle.

2.1.1 The Planar Lower Bound

Without loss of generality, we assume n is a multiple of 3. The adversary "presents" the following set of points:

$$S \triangleq \bigcup_{i=1}^{n/3} \{(-1, 4i), (0, 4i+1), (1, 4i)\}.$$

The set S consists of three smaller sets of points, evenly spaced along vertical line segments. See Figure 2.1(a). If we pick points p and r from the left and right segments, respectively, there is a unique point q in the middle segment such that the vertical distance from q to $\overleftrightarrow{pr}$ is exactly one. We shall refer to each such triple $\{p, q, r\}$ as a *collapsible triangle*, for the following reason. Without loss of generality, let q lie below $\overleftrightarrow{pr}$. If we perturb the set by moving p and r down by $1/2$ and moving q up by $1/2$, then the three points become colinear. See Figure 2.1(b). No other degeneracies are introduced by this perturbation; moreover, no other triangle changes orientation.

The adversary's point set S contains $n^2/9 = \Omega(n^2)$ collapsible triangles. If the algorithm does not check the orientation of every collapsible triangle, the adversary perturbs the set so that some unchecked triangle becomes degenerate. The algorithm cannot distinguish between the original point set and the perturbed point set. This completes the proof in the planar case.

It may be helpful to see what this construction looks like in the dual setting. Here

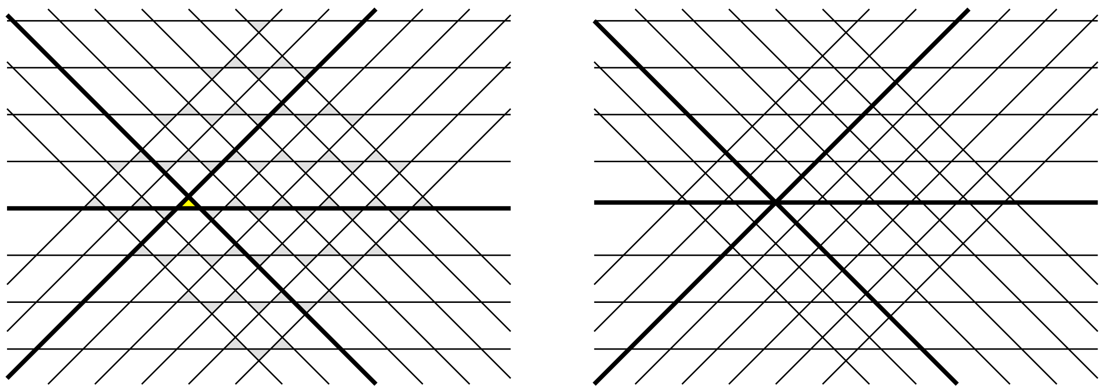


Figure 2.2. The dual version of our planar adversary construction. (a) The initial set, with collapsible triangles shaded. (b) The perturbed set, showing collapsed triangle.

we are given n lines in the plane and asked if any three of them have a common intersection.[3] The dual of the adversary's point set consists of three bundles of parallel lines. Two of the bundles meet in a mesh of squares, and the third cuts through the squares at a $45°$ angle, so that each square in the mesh has a small triangle cut off one corner. See Figure 2.2(a). Each of the small triangles in the mesh corresponds to a collapsible triangle in the primal point set. To collapse a triangle, the adversary simply translates its three bounding lines so that they intersect at the triangle's centroid. See Figure 2.2(b).

2.1.2 Higher Dimensions

For the d-dimensional problem, the adversary's point set consists of $d+1$ smaller sets. The points in each smaller set are evenly spaced along vertical line segments $l_0, l_1, \ldots, l_d$. These line segments intersect any horizontal hyperplane at the centroid and vertices of a regular $(d-1)$-simplex.

Without loss of generality, we assume n is a multiple of $3d$. Each of the outer segments $l_1, \ldots, l_d$ contains $2n/3d$ points, and l_0 contains the remaining $n/3$ points. The x_d coordinates of the outer points are multiples of $2d$ between 0 and $4n/3 - 2d$. Thus, any hyperplane defined by d points, one from each outer segment, intersects the x_d-axis at an even integer coordinate between 0 and $4n/3 - 2d$. The points in the inner set lie at alternate odd integer coordinates between 1 and $4n/3 + 1$. This gives us $\lfloor (d-1)/2 \rfloor$

[3]The restriction to nonvertical colinearities in the primal setting is reflected in the dual by ignoring the intersection points "at infinity" between parallel lines.

"wasted" points at the top of the inner segment, which we can ignore.

Suppose we pick one point from each of the outer sets. These points define a hyperplane h. The vertical distance between h and the unique point in the inner set that is closest to h is exactly 1. We refer to each such set of $d+1$ points as a *collapsible simplex*. The adversary can make any collapsible simplex degenerate by simultaneously moving the inner point up and the outer points down (or vice versa) a distance of $1/2$. Clearly, no other simplex changes orientation because of this perturbation. There are $(2n/3d)^d = \Omega(n^d)$ collapsible simplices in the adversary's point set, each of which must be checked by the algorithm.

This completes the proof of Theorem 2.1.

Since collapsing a simplex changes the order type of the set, we immediately have the following corollary.

Corollary 2.2. *Any decision tree that determines the order type of a set of n points in $\mathbb{R}^d$, using only sidedness queries, must have depth $\Omega(n^d)$.*

2.1.3 Beating the Lower Bound

If we know *in advance* that the points lie on $d+1$ vertical lines, then our $\Omega(n^d)$ lower bound can be defeated for all $d > 2$. In this special case, we can detect nonvertical degeneracies in $O(n^{d/2})$ time if d is even, and in $O(n^{(d+1)/2} \log n)$ time if d is odd. The algorithms that achieve these running times do not use only sidedness queries, but also compute the signs of certain linear forms. In addition to providing a pedagogical example of the importance of choosing the right model of computation, these algorithms suggest that a new approach may be required to extend our lower bounds into more general models of computation, at least in higher dimensions.

Suppose we are given $d+1$ sets $S_0, S_1, \ldots, S_d \subset \mathbb{R}^d$, each containing n points, such that each set S_i is contained in a vertical line l_i. The only possible nonvertical degeneracies contain one point from each line. The positions of the lines l_i determine constants a_i such that points $p_0 \in l_0, \ldots, p_d \in l_d$ lie on a nonvertical hyperplane if and only if their dth coordinates satisfy the equation

$$\sum_{i=0}^{d} a_i p_{id} = 0.$$

We describe two algorithms, one for even dimensions and one for odd dimensions. Our algorithms compare the weighted sums of tuples of dth coordinates of points, where the weight of each point is determined by the set from which it is taken. We call such a query a *tuple comparison*. Both algorithms work in two phases, a sorting phase and a scanning phase. In the sorting phase, both algorithms perform $\lceil d/2 \rceil$-tuple comparisons. In the scanning phase, both algorithms perform sidedness queries. In the odd-dimensional case, the sidedness queries we perform are actually $(d+1)/2$-tuple comparisons. In the discussion that follows, p_i always refers to a point in S_i.

If d is even, we sort all possible values of the expressions

$$\sum_{i=0}^{d/2-1} a_i p_{id} \quad \text{and} \quad \sum_{i=d/2}^{d-1} a_i p_{id}.$$

Then for each point $p_d \in S_d$, we scan through the two lists, looking for a pair of elements whose sum is $-a_d p_{dd}$. This algorithm runs in $O(n^{d/2+1})$ time.

If d is odd, we sort all possible values of the expressions

$$\sum_{i=0}^{\lfloor d/2 \rfloor} a_i p_{id} \quad \text{and} \quad \sum_{i=\lceil d/2 \rceil}^{d} -a_i p_{id},$$

and then simultaneously scan through the two lists for duplicate elements. This algorithm runs in $O(n^{(d+1)/2} \log n)$ time.

A simple variant of the odd-dimensional algorithm can be used to solve a slightly more general problem, in which the points are only constrained to lie on two vertical $(d+1)/2$-flats, which necessarily intersect at a vertical line l. Instead of sorting weighted sums, we sort the possible positions at which the affine hulls of $(d+1)/2$-tuples of points from the same $(d+1)/2$-flat intersect l. This algorithm also runs in $O(n^{(d+1)/2} \log n)$ time.

The special case of the affine degeneracy problem solved by these algorithms is an example of a *linear satisfiability problem*: Given a set of n real numbers, does any subset satisfy a fixed linear equation? We will consider linear satisfiability problems in greater detail in Chapter 5. The main result of that chapter (Theorem 5.1) implies that the algorithms we have just described are optimal, except possibly for a logarithmic factor when d is odd, when only sidedness queries and $\lceil d/2 \rceil$-tuple comparisons are allowed.

2.2 Lower Bounds for The General Problem

The *weird moment curve*, denoted $\omega_d(t)$, is the parameterized curve

$$\omega_d(t) = \left(t, t^2, \ldots, t^{d-1}, t^{d+1}\right).$$

where the parameter t ranges over the reals. The weird moment curve is similar to the standard moment curve $(t, t^2, \ldots, t^{d-1}, t^d)$, except that the degree of the last coordinate is increased by one.

If we project the weird moment curve down a dimension by dropping the last coordinate, we get a standard moment curve. Since every set of points on the standard moment curve is affinely nondegenerate, no d points on the d-dimensional weird moment curve lie on a single $(d-2)$-flat. However, it is possible for $d+1$ points to all lie on a single hyperplane. The following lemma characterizes these affine degeneracies.

Lemma 2.3. *Let $x_0 < x_1 < \cdots < x_n$ be real numbers. The orientation of the simplex $(\omega_d(x_0), \omega_d(x_1), \ldots, \omega_d(x_d))$ is given by the sign of $\sum_{i=0}^{d} x_i$. In particular, the simplex is degenerate if and only if $\sum_{i=0}^{d} x_i = 0$.*

Proof: The orientation of the simplex $(\omega_d(x_0), \omega_d(x_1), \ldots, \omega_d(x_d))$ is given by the sign of the determinant of the following matrix.

$$M = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^{d-1} & x_0^{d+1} \\ 1 & x_1 & x_1^2 & \cdots & x_1^{d-1} & x_1^{d+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_d & x_d^2 & \cdots & x_d^{d-1} & x_d^{d+1} \end{bmatrix}$$

The determinant of M is an antisymmetric polynomial of degree $\binom{d+1}{2} + 1$ in the variables x_i, and it is divisible by $(x_i - x_j)$ for all $i < j$. It follows that

$$\frac{\det M}{\prod_{i<j}(x_j - x_i)}$$

is a symmetric polynomial of degree one, and we easily observe that its leading coefficient is 1. (This polynomial is well-defined, since the x_i are distinct.) The only such polynomial is $\sum_{i=0}^{d} x_i$. □

This result, or at least its proof, is hardly new. If we replace the weird moment curve by any polynomial curve, the orientation of a simplex is given by the sign of a Schur polynomial [131]. A determinantal formula for Schur polynomials was discovered by Jacobi in the mid-1800's [100].

Theorem 2.4. *Any decision tree that decides whether a set of* n *points in* $\mathbb{R}^d$ *is affinely degenerate, using only sidedness queries, must have depth* $\Omega(n^d)$. *If* $d \geq 3$, *this lower bound holds even when the points are known in advance to be in convex position.*

Proof: Let X denote the set of integers from $-dn$ to n. If we lift X up to the weird moment curve, the resulting set of points $\omega_d(X)$ contains $\Omega(n^d)$ degenerate simplices. To pick a degenerate simplex, choose arbitrary distinct positive elements $x_1, x_2, \ldots, x_d \in X$, and let $x_0 = -\sum_i x_i$.

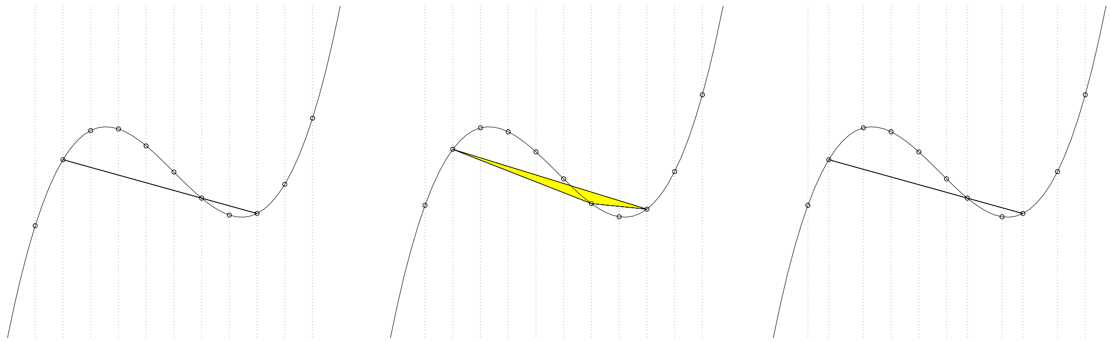
The adversary initially presents the point set $\omega_d(X')$, where X' denotes the set $X+1/(2d+2) = \{x+1/(2d+2) \mid x \in X\}$. This point set is affinely nondegenerate, since the the sum of any $d+1$ elements in X' is always a half-integer. Choose arbitrary distinct positive elements $x'_1, x'_2, \ldots, x'_d \in X'$, and let $x'_0 = 1/2 - \sum_i x'_i$. The points $\omega_d(x'_i)$ form a collapsible simplex. To collapse it, the adversary shifts the points back to their original positions $\omega_d(x_i)$. The collapsed simplex is obviously degenerate. Moreover, since the expression $\sum_{i=0}^{d} x'_i$ changes by at most $1/2 - 1/(2d+2) < 1/2$ for any other simplex, no other simplex changes orientation. In particular, the collapsed simplex is the only degenerate simplex.

The adversary's point set contains $\binom{n}{d} = \Omega(n^d)$ collapsible simplices. If an algorithm does not check the orientation of every collapsible simplex, then the adversary perturbs the input so that some unchecked simplex becomes degenerate. The algorithm cannot distinguish between the original point set and the perturbed point set, even though only one of them is degenerate.

Since every set of points on the standard moment curve is in convex position, every set of points on the d-dimensional weird moment curve in convex position if $d \geq 3$. (Given a set of points in convex position in the plane, we can easily determine whether any three are colinear in $O(n \log n)$ time.) □

Figure 2.3 illustrates the new two-dimensional construction. In order to make the colinearities more visible, the figure uses a curve of the form $y = x^3 - \alpha x$; since this is a linear transformation of the unit cubic $y = x^3$, all colinearities are preserved.

We emphasize that if the points are known *in advance* to lie on the weird moment curve, affine degeneracies can be detected in $O(n^{d/2})$ time if d is even, and in



(a) (b) (c)

Figure 2.3. A planar adversary construction for arbitrary degeneracies. (a) The degenerate configuration, with one degenerate triangle emphasized. (b) The adversary configuration, with the corresponding collapsible triangle. (c) The corresponding collapsed configuration.

$O(n^{(d+1)/2} \log n)$ time if d is odd, by simple algorithms that use more complicated queries, similar to the algorithms described in Section 2.1.3.

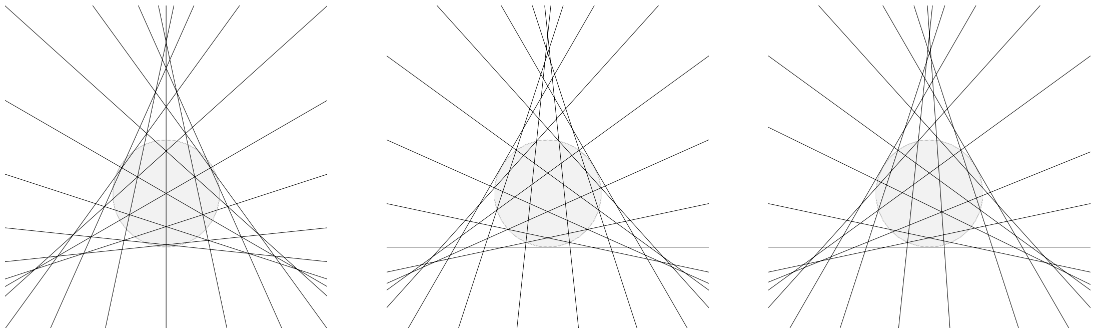
2.2.1 An Alternate Proof in Two Dimensions

In the 1950's, Sylvester noted that a set of n integer points on the unit cubic can have $n^2/8$ collinear triples [96]. Füredi and Palásti [84] improve this lower bound to roughly $n^2/6$ using a slightly different construction, which we describe below. We can use their construction to slightly improve our lower bound for the two-dimensional affine degeneracy problem. The resulting lower bound is the best that can be derived using our techniques, except possibly for some lower-order terms.

Füredi and Palásti describe their construction in the dual. Let $L(\alpha)$ be the line passing through the point $(\cos\alpha, \sin\alpha)$ at angle $-\alpha/2$ to the x-axis. The line $L(\alpha)$ also passes through the point $(\cos(\pi-2\alpha), \sin(\pi-2\alpha))$; if this is the same point as $(\cos\alpha, \sin\alpha)$, then the line is tangent to the unit circle at that point. Three lines $L(\alpha), L(\beta), L(\gamma)$ are concurrent if and only if $\alpha+\beta+\gamma \equiv 0 \pmod{2\pi}$. It follows that the set of lines $\{L(2\pi i/n) \mid 1 \le i \le n\}$ has $1 + \lfloor n(n-3)/6 \rfloor$ concurrent triples. See Figure 2.4(a). See [84] for further details. Related results are described in [24] and [72].

The set of lines $\{L((2i-1)\pi/n) \mid 1 \le i \le n\}$ has no concurrent triples, but its arrangement has $\lceil n(n-3)/3 \rceil$ triangular cells, each bounded by a triple of lines of the form

$$L((2i-1)\pi/n),\ L((2j-1)\pi/n),\ L((2k-1)\pi/n),$$



(a) (b) (c)

Figure 2.4. Another planar adversary construction for arbitrary degeneracies, following a construction of Füredi and Palásti. (a) The degenerate configuration. (b) The adversary configuration. (c) A collapsed configuration.

where $i+j+k \equiv 1$ or $2 \pmod{n}$. See Figure 2.4(b). Each of these triangles is collapsible; to collapse such a triangle, we shift each of its three defining lines by $\pi/3n$, resulting in the lines

$$L((2i-2/3)\pi/n),\ L((2j-2/3)\pi/n),\ L((2k-2/3)\pi/n),$$

if $i+j+k \equiv 1 \pmod{n}$, or

$$L((2i-4/3)\pi/n),\ L((2j-4/3)\pi/n),\ L((2k-4/3)\pi/n),$$

if $i+j+k \equiv 2 \pmod{n}$. See Figure 2.4(c). We easily verify that the collapsed triangle is degenerate, and that no other triangle changes orientation, since the sum of any other triple of defining angles changes by at most $2\pi/3n < \pi/n$.

Theorem 2.5. *Any decision tree that decides whether a set of n points in $\mathbb{R}^2$ is affinely degenerate, using only sidedness queries, must have depth at least $\lceil n(n-3)/3 \rceil$.*

Grünbaum [96] proved that a simple arrangement of n lines in the projective plane can have at most $\lfloor n(n-1)/3 \rfloor$ triangular cells if n is even, and at most $\lfloor n(n-2)/3 \rfloor$ if n is odd. Thus, we cannot hope to prove a lower bound bigger than $n^2/3 + O(n)$ using collapsible triangles.

2.3 Allowable Queries

In this section, we identify a general class of computational primitives which, if added to our model of computation, do not affect our lower bounds. In fact, even if we allow any finite number of these primitives to be performed at no cost, the number of required sidedness queries is the same. These primitives include comparisons between coordinates of input points in any number of directions, comparisons between coordinates of hyperplanes defined by d-tuples of points, and in-sphere queries.

The model of computation we consider is a restriction of the algebraic decision tree model. Recall that in this model, the result of every query is given by the sign of a multivariate *query polynomial*, evaluated at the coordinates of the input. If the sign is zero (resp. nonzero), we say that the input is *degenerate* (resp. *nondegenerate*) with respect to that query. For example, a set of points is affinely degenerate if and only if it is degenerate with respect to some sidedness query.

A *projective transformation* of $\mathbb{R}^d$ (or more properly, of the projective space $\mathbb{RP}^d$) is any map that takes hyperplanes to hyperplanes. If we represent the points of $\mathbb{R}^d$ in homogeneous coordinates, a projective transformation is equivalent to a linear transformation of $\mathbb{R}^{d+1}$.

Let $X = \{-dn, 1 - dn, \ldots, n\}$ be the set of numbers described in the proof of Theorem 2.4. We call an algebraic query *allowable* if for some projective transformation ϕ, the point configuration $\phi(\omega_d(X))$ is nondegenerate with respect to that query. Our choice of terminology is justified by the following theorem.

Theorem 2.6. *Any decision tree that decides whether a set of* n *points in* $\mathbb{R}^d$ *is affinely degenerate, using only sidedness queries and a finite number of allowable queries, requires* $\Omega(n^d)$ *sidedness queries in the worst case. If* $d \geq 3$*, this lower bound holds even when the points are known in advance to be in convex position.*

Proof: Every d-dimensional projective transformation can be written as a $(d+1) \times (d+1)$ real matrix. For any polynomial q, the set of projective tranformations ϕ such that $q(\phi(\omega_d(X))) = 0$ is an algebraic variety in $\mathbb{R}^{(d+1)\times(d+1)}$. It follows that if *some* projective transformation makes $\omega_d(X)$ nondegenerate with respect to an algebraic query, then *almost every* projective transformation (*i.e.*, all but a measure-zero subset) makes $\omega_d(X)$ nondegenerate. Moreover, for any finite set of allowable queries, almost every projective transformation makes $\omega_d(X)$ nondegenerate with respect to all of them. Let ϕ be such a transformation.

Now consider the degenerate configuration $\phi(\omega_d(X))$ as a single point in the configuration space $\mathbb{R}^{dn}$. Every algebraic query induces an algebraic surface in this space, consisting of all configurations that are degenerate with respect to that query. Since algebraic surfaces are closed, if $\phi(\omega_d(X))$ is nondegenerate with respect to some finite set of alllowable queries, then for all X' in an open neighborhood of X in $\mathbb{R}^n$, the configuration $\phi(\omega(X'))$ is also nondegenerate with respect to that set of queries.

The theorem now follows from a slight modification of the proof of Theorem 2.4. Let $\varepsilon > 0$ be some sufficiently small real number. The set $\phi(\omega_d(X+\varepsilon))$ is affinely nondegenerate, but has $\Omega(n^d)$ collapsible simplices, each corresponding to a degenerate simplex in $\phi(\omega_d(X))$. No allowable query can distinguish between $\phi(\omega_d(X+\varepsilon))$ and any collapsed configuration, or even between $\phi(\omega_d(X+\varepsilon))$ and $\phi(\omega_d(X))$. □

We give below a (nonexhaustive!) list of allowable queries. We leave the proofs that these queries are in fact allowable as easy exercises.

- Comparisons between points in any fixed direction are allowable. In fact, we can allow the input points to be presorted in any finite number of fixed directions. A similar result was described by Seidel in the context of three-dimensional convex hull lower bounds [133, Theorem 5]. We emphasize that the directions in which these comparisons are made must be fixed in advance. No matter how we transform the adversary configuration, there is always *some* direction in which a point comparison can distinguish it from a collapsed configuration.

- More generally, deciding which of two points is hit first by a hyperplane rotating around a fixed $(d-2)$-flat is allowable. We can even presort the points by their cyclic orders around any finite number of fixed $(d-2)$-flats. If the $(d-2)$-flat is "at infinity", then "rotation" is just translation, and we have the previous notion of point comparison. We can interpret this type of query in dual space as a comparison between the intersections of two hyperplanes with a fixed line. Again, we emphasize that the $(d-2)$-flats must be fixed in advance.

- Sidedness queries in any fixed lower-dimensional projection are allowable. This is a natural generalization of point comparisons, which can be considered sidedness queries in a one-dimensional projection. We can even specify in advance the complete order types of the projections onto any finite number of fixed affine subspaces. (As a technical point, we would not actually include this information as part of the input, since this would drastically increase the input size. Instead, knowledge of the projected order types would be hard-wired into the algorithm.)

- "Second-order" comparisons between vertices of the dual hyperplane arrangement, in any fixed direction, are also allowable. Such a query can be interpreted in the primal space as a comparison between the intersections of two hyperplanes, each defined by a d-tuple of input points, with a fixed line. To prove that such a query is allowable, it suffices to observe that a projective transformation of the primal space induces a projective transformation of the dual space, and vice versa. Note that a second-order comparison is algebraically equivalent to a sidedness query if the two d-tuples share $d-1$ points.

- Since most projective transformations do not map spheres to spheres, in-sphere queries are allowable. Given $d+2$ points, an in-sphere query asks whether the first point lies "inside", on, or "outside" the oriented sphere determined by the other $d+1$ points. (See Chapter 4.) Similarly, in-sphere queries in any fixed lower-dimensional projection are allowable.

- Distance comparisons between pairs of points or pairs of projected points are allowable. More generally, comparing the measures of pairs of simplices of dimension less than d — for example, comparing the areas of two triangles when $d > 2$ — defined either by the original points or by any fixed projection, are allowable.

On the other hand, comparing the volumes of arbitrary simplices of *full* dimension is *not* allowable. In any projective transformation of $\omega_d(X)$, all of the degenerate simplices have the same (zero) volume. It is not possible to collapse a simplex in any adversary configuration while maintaining the order of the volumes of the other collapsible simplices.

2.4 Implications and Open Problems

A problem similar to finding degeneracies is finding the minimum measure simplex. Unfortunately, our results are *not* sufficient to improve the $\Omega(n \log n)$ lower bound on this problem. Any algorithm that finds the minimum measure simplex must be able to compare the values of arbitrary sidedness determinants, and such comparisons are not allowed in

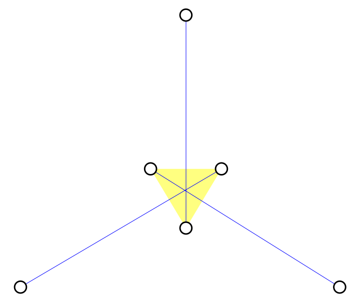


Figure 2.5. Minimum area triangles are not necessarily collapsible.

any model of computation in which our lower bounds hold. This difference may be best understood by looking at the one-dimensional case: $\Omega(n \log n)$ comparisons are required to sort a list of n numbers, but a stronger model is required to say anything about finding the closest pair. It seems impossible to apply our "collapsible simplex" argument in a model that allows comparisons between simplex volumes; a radically new idea is called for. We quickly note that the minimum measure simplex is not necessarily collapsible; see Figure 2.5.

The planar affine degeneracy problem is an example of what Gajentaan and Overmars [86] call *3sum-hard problems*.[4] Formally, a problem is 3sum-hard if the following problem can be reduced to it in subquadratic time:

3sum: Given a set of real numbers, do any three sum to zero?

Thus, a subquadratic algorithm for any 3sum-hard problem would imply a subquadratic algorithm for 3sum, and a sufficiently powerful quadratic lower bound for 3sum would imply similar lower bounds for every 3sum-hard problem. Examples of 3sum-hard problems include several degeneracy detection, separation, hidden surface removal, and motion planning problems in two and three dimensions. Gajentaan and Overmars [86] show that the planar affine degeneracy problem is 3sum-hard, by considering a lifting from the reals to the unit cubic.[5] In fact, the restricted problem considered in Section 2.1.1 is equivalent to 3sum since there are simple linear-time reductions in both directions. Our results imply a quadratic lower bound for 3sum; we will present further details in Chapter 5.

Given these reductions, one might think that we have just proven that every

[4] Some earlier papers, including [73], used the more suggestive but potentially misleading term "n^2-hard" [85] (but see [20]).

[5] This observation was the initial inspiration for my "weird moment curve" argument.

3SUM-hard problem requires $\Omega(n^2)$ time. Unfortunately, this is not the case. Many of the reductions discussed in [86] require primitives that our models of computation do not allow. In these cases, one may still be able to achieve quadratic lower bounds by directly applying the techniques in this chapter. For example, consider the following problem, which Gajentaan and Overmars call SEPARATOR2: Given a set of n non-intersecting line segments in the plane, is there a line that separates the set into two non-empty subsets? Using the techniques in this paper, one can derive a quadratic lower bound for this problem, under a model that allows sidedness queries and allowable queries among the endpoints of the segments.

Even so, some 3SUM-hard problems, like the minimum-area triangle problem, cannot be solved in the models in which our techniques apply. Many of these problems already have $O(n^2)$ solutions that use primitives outside our models.

In light of these shortcomings, an obvious open question is whether our lower bound also holds in models where even more queries are allowed. Ultimately, of course, we would like a lower bound that holds in a general model of computation such as algebraic decision trees, but this seems to be completely out of reach.

The other possibility, of course, is that there is a subquadratic algorithm in some completely different model of computation. The situation may be comparable to sorting or element uniqueness—$\Omega(n \log n)$ time is required to sort using algebraic decision trees [16], but there are significantly faster sorting algorithms in integer RAM models [83, 9].

Are there faster algorithms for useful special cases? For example, a set of n points in the plane in (loosely) convex position has only n collapsible triangles, and we can easily detect colinear triples in such a set in $O(n \log n)$ time. Is there a "structure-sensitive" algorithm for detecting affine degeneracies, whose running time depends favorably on the number of collapsible triangles? Such an algorithm might be useful for solving real-world instances of other 3SUM-hard problems such as planar motion planning and hidden surface removal [86].

2.5 Out on a Limb

At the risk of annoying the reader, let me close this chapter by outlining some more evidence that the three-colinear-points problem "really" requires $\Omega(n^2)$ time. Readers looking for more theorems will be disappointed; my aim is only to provide some intuition and hopefully provoke further research.

An arrangement of *pseudolines* is a collection of curves in the plane, each homeomorphic to a straight line, such that any pair intersect transversely in exactly one point. Such an arrangement is *simple* if no three pseudolines pass through a single point. An pseudoline arrangement is *stretchable* if it can be continuously deformed into an arrangement of straight lines. A theorem of Mnëv [116] (see also [137, 97, 128]) implies that determining if a pseudoline arrangement is stretchable is NP-hard.[6]

Every known algorithm that detects degeneracies in arrangements of lines [39, 65, 68, 69] can also be used to detect degeneracies in arrangements of pseudolines. In fact, there is no known algorithmic separation of lines and pseudolines. That is, there is no known problem that can sensibly be asked about both lines and pseudolines (for example, "Sort the intersection points." or "How many edges are in the kth level?"), such that an efficient algorithm is known for the straight line version that doesn't also work for the pseudoline version. In light of Mnëv's theorem, this is perhaps not terribly surprising. (A relevant pseudo-algorithmic result is Steiger and Streinu's proof that any decision tree that sorts the intersection points of a pseudoline arrangement must have depth $\Omega(n^2 \log n)$, but the vertices of a line arrangement can be sorted by a *nonuniform* algorithm that uses only $O(n^2)$ comparisons [139]. See Chapter 5 for further discussion of nonuniform algorithms.)

There are $2^{\Theta(n^2)}$ combinatorially distinct arrangements of n pseudolines in the plane [105, 79]. (Recall from the beginning of this chapter that only $n^{\Theta(n)}$ of these are stretchable [89].) It follows immediately that determining the order type of a pseudoline arrangement requires $\Omega(n^2)$ time. Moreover, since *every* triangular cell in a pseudoline arrangement can be "flipped" to produce a new pseudoline arrangement and there are arrangements with $\Omega(n^2)$ triangular cells, $\Omega(n^2)$ sidedness queries are necessary to decide if a pseudoline arrangement is simple.

These observations suggest that deciding if a line arrangement is simple requires $\Omega(n^2)$ time because (1) deciding if a *pseudo*line arrangement is simple requires quadratic time, and (2) it is not possible for an efficient algorithm to know that its input consists of straight lines and not arbitrary pseudolines (unless, perhaps, P=NP). This suggestion is far too vague to call a "conjecture"; in particular, I haven't mentioned a specific model of com-

[6] Mnëv showed any primary semialgebraic set defined over the integers is stably homotopy-equivalent to the realization space of some rank-3 oriented matroid (*i.e.*, some pseudoline arrangement). See [18] for further implications of this remarkable result.

putation. Nevertheless, any results in this direction (even just formalizing the "conjecture") would be interesting.

[items 1–19 omitted]
20. I have no arguments to offer, my figures are my proofs.
21. The laws of nature are in harmony with me and sustain me.
22. Laugh away these facts and truths if you can.

— Carl Theodore Heisel, *The Circle Squared Beyond Refutation*, 1934