30 POLYGONS

Joseph O'Rourke, Subhash Suri, and Csaba D. Tóth

INTRODUCTION

Polygons are one of the fundamental building blocks in geometric modeling, and they are used to represent a wide variety of shapes and figures in computer graphics, vision, pattern recognition, robotics, and other computational fields. By a polygon we mean a region of the plane enclosed by a simple cycle of straight line segments; a ***simple cycle*** means that nonadjacent segments do not intersect and two adjacent segments intersect only at their common endpoint. This chapter describes a collection of results on polygons with both combinatorial and algorithmic flavors. After classifying polygons in the opening section, Section 30.2 looks at simple polygonizations, Section 30.3 covers polygon decomposition, and Section 30.4 polygon intersection. Sections 30.5 addresses polygon containment problems and Section 30.6 touches upon a few miscellaneous problems and results.

30.1 POLYGON CLASSIFICATION

Polygons can be classified in several different ways depending on their domain of application. In chip-masking applications, for instance, the most commonly used polygons have their sides parallel to the coordinate axes.

GLOSSARY

Simple polygon: A closed region of the plane enclosed by a simple cycle of straight line segments.

Convex polygon: The line segment joining any two points of the polygon lies within the polygon.

Monotone polygon: Any line orthogonal to the direction of monotonicity intersects the polygon in a single connected piece.

Star-shaped polygon: The entire polygon is visible from some point inside the polygon.

Orthogonal polygon: A polygon with sides parallel to the (orthogonal) coordinate axes. Sometimes called a ***rectilinear polygon***.

Orthogonally convex polygon: An orthogonal polygon that is both x- and y-monotone.

Polygonal chain: A sequence of connected, non-self-intersecting line segments forming a subportion of a simple polygon's boundary. A chain is convex or reflex if all internal angles are convex or reflex respectively.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry, J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

Spiral polygon: A polygon bounded by one convex chain and one reflex chain.

Crescent polygon: A monotone spiral polygon.

Pseudotriangle: A polygon with exactly three convex angles. Each is pair of convex vertices is connected either by a single segment or a reflex chain.

Histogram polygon: An orthogonal polygon bounded by an x-monotone polygonal chain and a single horizontal line segment.

POLYGON TYPES

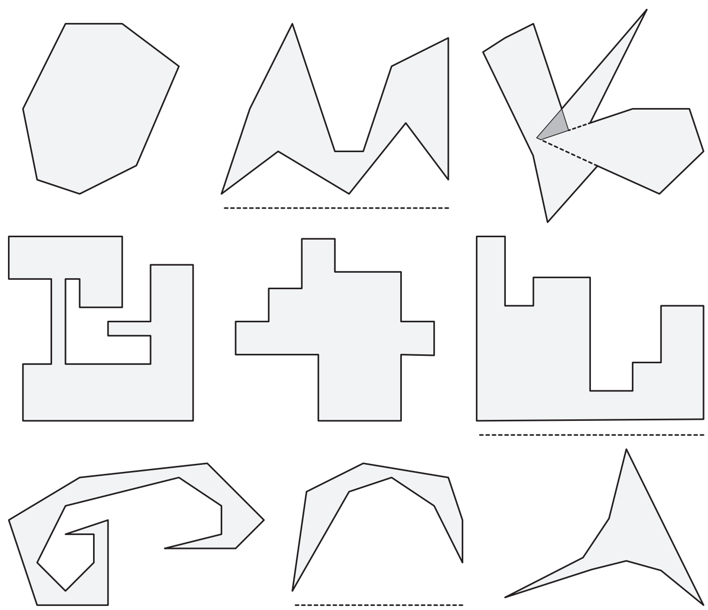

FIGURE 30.1.1
Various varieties of polygons: convex, monotone, star-shaped (with kernel), orthogonal, orthogonally convex, histogram, spiral, crescent, and pseudotriangle.

Before starting our discussion on problems and results concerning polygons, we clarify a few technical issues. The qualifier "simple" in the definition of a simple polygon states a *topological* property, meaning "nonself-intersection." Not to be confused with "uncomplicated polygons," in fact, these polygons include the most complex among polygons that are topologically equivalent to a disk (see the classification below). Finally, we will make a standard ***general position*** assumption throughout this chapter that no three vertices of a polygon are collinear.

The relationship between several classes of polygons can be understood using the concept of visibility (see Chapter 33). We say that two points x and y in a polygon P are mutually ***visible*** if the line segment $\overline{xy}$ does not intersect the complement of P; thus the segment $\overline{xy}$ is allowed to graze the polygon boundary but not cross it. We call a set of points $K \subset P$ the ***kernel*** of P if all points of P are visible from every point in the kernel. Then, a polygon P is convex if $K = P$; the polygon is star-shaped if $K \neq \emptyset$ (see the star-shaped polygon in Fig. 30.1.1);

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry, J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

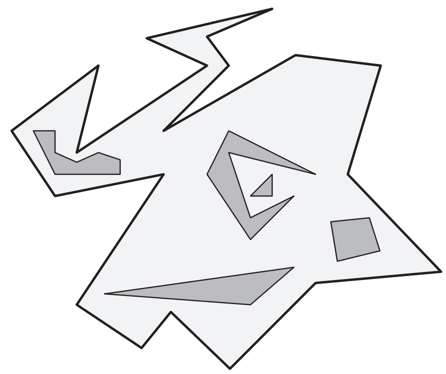

FIGURE 30.1.2
A polygon with five holes.

otherwise, the polygon is merely a simple polygon. Speaking somewhat loosely, a monotone polygon can be viewed as a special case of a star-shaped polygon with the exterior kernel at infinity—that is, a monotone polygon can be decomposed into two polygonal chains, each of which is entirely visible from the (same) point at infinity in the extended plane. A pseudotriangle is often but not always star-shaped.

The kernel has been generalized to “left-” and “right-kernels,” whose definition we leave to [TTW11].

By definition, a simple polygon P is a polygon *without holes*—that is, the interior of the polygon is topologically equivalent to a disk. A ***polygon with holes*** is a higher-genus variant of a simple polygon, obtained by removing a nonoverlapping set of strictly interior, simple subpolygons from P. Figure 30.1.2 illustrates the distinction between a simple polygon and a polygon with holes.

An important class of polygons are the *orthogonal polygons*, where all edges are parallel to the coordinate axes. These polygons arise quite naturally in industrial applications, and often algorithms are faster on these more structured polygons.

30.2 SIMPLE POLYGONIZATIONS

It would be useful to have a clear notion of a “random polygon” so that algorithms could be tested for typical rather than worst-case behavior. This leads to the issue of generating the ***simple polygonalizations*** of a fixed point set S, a simple polygon whose vertices are precisely the points of S. That every set S of n points in general position has a simple polygonization has been known since Steinhaus in 1964. In fact, every such S has a star-shaped polygonization (Graham), a monotone polygonization (Grünbaum), and a spiral polygonization. See [IM11] for the latter result and a survey of earlier work.

The “space” of all polygonizations of a fixed point set S can be explored through two elementary moves, which is one approach to generating random polygons: generate a random S, then walk between polygonizations to a “random” polygonization in the space [DFOR10]. The size of this space can be exponential.

Considerable research has focused on quantifying this exponential, that is, counting the maximum number of polygonizations over all n-point sets S. (Of course, the minimum is 1 because vertices of a convex polygon can form only one simple polygon.) Current best bounds for the maximum are $\Omega(4.64^n)$ [GNT00] and $O(54.55^n)$ [SSW13].

Subsets of the full space have been explored, in particular, finding polygonizations with special properties. The shortest perimeter polygonization is the Euclidean TSP; see Chapter 31. There is a 2π approximation (under certain conditions) for the longest perimeter polygonization [DT10]. Finding the minimum or maximum area polygonization is NP-hard [Fek00]. Sufficient conditions have been found for the existence of a polygonization of n red points while enclosing [HMO$^+$09] or excluding [FKMU13] a set of blue points.

Another goal has been to minimize the number of reflex vertices in a polygonization of a point set S. This minimum is known as the *reflexivity* of S. A tight upper bound is known as a function of n_I, the number of points of S strictly interior to the convex hull: the worst case is $\lceil n_I/2 \rceil$, and this can be achieved by a class of examples [AFH$^+$03]. As a function of $n = |S|$, the best bound is $\frac{5}{12}n + O(1) \approx 0.4167n$ [AAK09]. Another criterion in the same spirit is to minimize the sum of the "turn angles" at reflex interior angles [Ror14]. The worst case is conjectured to be $2\pi(1 - 1/(n-1))$, but only $2\pi - \pi/((n-1)(n-3))$ is proved.

OPEN PROBLEMS

1. *Simple polygonalization*: Can the number of simple polygonalizations of a set of n points in the plane be computed in polynomial time?
2. *Partial polygons*: Is there a polynomial time algorithm that can decide, for a set of n points and a set of edges among them, whether there is a polygonization that uses all the given edges?

30.3 POLYGON DECOMPOSITION

Many computational geometry algorithms that operate on polygons first decompose them into more elementary pieces, such as triangles or quadrilaterals. There is a substantial body of literature in computational geometry on this subject.

GLOSSARY

Steiner point: A vertex not part of the input set.

Diagonal: A line segment connecting two polygon nonadjacent vertices and contained in the polygon. An ***edge*** connects adjacent vertices.

Polygon cover: A collection of subpolygons whose union is exactly the input polygon.

Polygon partition: A collection of subpolygons with *pairwise disjoint* interiors whose union is exactly the input polygon.

Dissection: A dissection of one polygon P to another Q is a partition of P into a finite number of pieces that may be reassembled to form Q.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry,
J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

Decompositions may be classified along two primary dimensions: covers or partitions, and with or without Steiner points. A cover permits a polygon in the shape of the symbol "+" to be represented as the union of two rectangles, whereas a minimal partition requires three rectangles, a less natural decomposition. Decompositions without Steiner points use diagonals, and are in general easier to find but less parsimonious. For each of the four types of decomposition, different primitives may be considered. The ones most commonly used are rectangles, convex polygons, star-shaped polygons, spiral polygons, and trapezoids. Restrictions on the shape of the piece being decomposed are often available; for example, orthogonal polygons for rectangle covers. Lastly, the distinction between simple polygons and polygons with holes is often relevant for algorithms. The most celebrated polygon partition problem is the "polygon triangulation problem."

TRIANGULATION ALGORITHMS

The polygon triangulation problem is to dissect a polygon into triangles by drawing a maximal number of noncrossing diagonals. Only the vertices of the polygon are used as triangle vertices, and no additional interior (Steiner) points are allowed. It is an easy and well-known result that every simple polygon can be triangulated, and that the number of triangles is invariant over all triangulations. More precisely:

THEOREM 30.3.1

Every simple polygon admits a triangulation, and every triangulation of an n-vertex polygon has $n-3$ diagonals and $n-2$ triangles.

(But note that the natural generalization to $\mathbb{R}^3$—that every polyhedron admits a tetrahedralization—is false: see Chapter 29.) The number of possible diagonals in a polygon may vary from linear (e.g., a spiral polygon) to quadratic (e.g., a convex polygon). A diagonal that breaks the polygon into two roughly equal halves is called a ***balanced*** diagonal. In designing his $O(n \log n)$ time algorithm for triangulating a polygon, Chazelle [Cha82] proved the following fact, which has found numerous applications in divide-and-conquer based algorithms for polygons, e.g., to ray-shooting [HS95]:

THEOREM 30.3.2

Every n-vertex simple polygon admits a diagonal that breaks the polygon into two subpolygons, neither one with more than $\lceil 2n/3 \rceil + 1$ vertices.

By recursively dividing the polygon using balanced diagonals, we get a balanced decomposition of P, which can be modeled by a tree of height $O(\log n)$. The existence of a balanced diagonal follows easily once we consider the graph-theoretic dual of a triangulation. This dual graph of a polygon triangulation is a tree, with maximum node degree three. Diagonals of the triangulation correspond to the edges of the dual tree, and thus a balanced diagonal corresponds to an edge whose removal breaks the tree into two subtrees, each with at most $\lceil 2n/3 \rceil + 1$ nodes.

The problem of computing a triangulation of a polygon has had a long and distinguished history [O'R87], culminating in Chazelle's linear-time algorithm [Cha91]. Table 30.3.1 lists some of the best-known algorithms for this problem. The algorithm in [Sei91] is a randomized Las Vegas algorithm (see Chapter 44). All others are deterministic algorithms, with worst-case time bounds as shown.

TABLE 30.3.1 Results on triangulating a simple polygon.

TIME COMPLEXITY	ALGORITHM	SOURCE
$O(n \log n)$	monotone pieces	[GJPT78]
$O(n \log n)$	divide-and-conquer	[Cha82]
$O(n \log n)$	plane sweep	[HM85]
$O(n \log^\star n)$	randomized	[Sei91]
$O(n)$	polygon cutting	[Cha91]

Chazelle's deterministic linear-time algorithm is formidably complex, but has led to a simpler randomized algorithm that runs in linear expected time [AGR01].

If the polygon contains holes (Figure 30.1.2), then $\Theta(n \log n)$ time is both necessary and sufficient for triangulating the region [HM85]. See Table 30.3.2.

TABLE 30.3.2 Results on triangulating a polygon with holes.

TIME COMPLEXITY	ALGORITHM	SOURCES
$O(n \log n)$	plane sweep	[HM85]
$O(n \log n)$	local sweep	[RR94]

COUNTING TRIANGULATIONS

The number of triangulations of a simple polygon with n vertices is at least 1 (every polygon can be triangulated) and at most $C_{n-2} = \frac{1}{n-1}\binom{2n-4}{n-2}$, the $(n-2)$-th Catalan number (for convex polygons). The number of triangulations of a given polygon of n vertices can be determined in $O(n^3)$ time by dynamic programming [ES94]; or in $O(e^{3/2})$ time, where $e = O(n^2)$ is the number of diagonals of the polygon [DFH$^+$99], if the diagonals are given. In $O(n \log n)$ additional time, random triangulations (with uniform distribution) can be generated [DQTW05], improving on an earlier $O(n^4)$ algorithm [DFH$^+$99].

SPECIAL TRIANGULATIONS

Polygon triangulations with either minimum or maximum edge length can be found in $O(n^2)$ time via dynamic programming [Kli80].

Pseudotriangulations have many applications, for example, to motion planning, and to kinetic data structures. Every simple polygon admits a pseudotriangulation in which the maximum vertex-degree is at most 5, and this bound is the best possible [AHST03]. However, the dual graph of a minimum pseudotriangulation may have degree $\Omega(\log n)$.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry, J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

COVERS AND PARTITIONS

The problem of decomposing polygons into different types of simpler polygons has numerous applications within and outside computational geometry (see, e.g., Chapter 57). Unlike the triangulation problem, most variants of the covering and partitioning problems turn out to be provably hard. In a covering problem, the goal is to cover the interior of the polygon with the smallest number of subpolygons of a particular type, for instance, convex or star-shaped polygons. Table 30.3.3 lists results for various polygon covering problems. In this table, "cover type" refers to the family of polygons allowed in the cover, while "domain" refers to the polygonal region that needs to be covered. For the most part, we consider only four types of domains: simple polygons, with and without holes, and orthogonal polygons, with and without (orthogonal) holes. In all of these problems, the cover or partition pieces are allowed to use Steiner points for their vertices. Almost all variations of the covering problem are intractable. Even the minimum convex cover problem without Steiner points is NP-complete [Chr11].

TABLE 30.3.3 Results on polygon covering problems.

COVER TYPE	DOMAIN	HOLES	COMPLEXITY	SOURCE
Rectangles	orthogonal	Y	NP-complete	[Mas78]
Convex–star	polygons	Y	NP-hard	[OS83]
Star	polygons	N	NP-hard	[Agg84]
			APX-hard	[ESW01]
Rectangles	orthogonal	N	NP-hard	[CR94]
Squares	orthogonal	N	$O(n^{3/2})$	[ACKO88]
		Y	NP-complete	[ACKO88]
Convex	polygons	N	NP-hard	[CR94]
			APX-hard	[EW03]

The polygon-partitioning problems are similar to the covering problem, except that the tessellating pieces are not allowed to overlap: they have pairwise disjoint interiors. Table 30.3.4 collects results on polygon partitioning problems permitting Steiner points. Polynomial-time algorithms can be achieved for simple polygons using dynamic programming. The same problems, however, turn out to be intractable when the polygon has holes. Disallowing Steiner points also leads to polynomial-time algorithms. For example, partitioning a polygon without holes into the fewest convex pieces, not employing Steiner points, is achievable in $O(n^3 \log n)$ time [Kei85, KS02].

CONVEX COVER APPROXIMATIONS

The intractability of most covering and partitioning problems naturally leads to the question of approximability—how well can we approximate the size of an optimal cover or partition in polynomial time. In many cases, there are only a polynomial number of covering candidates—for instance, rectangle covers or convex poly-

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry,
J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

TABLE 30.3.4 Results on polygon partitioning problems.

PARTITION	DOMAIN	HOLES	COMPLEXITY	SOURCE
Convex	polygons	N	$O(n^3)$	[CD79]
Convex	polygons	Y	NP-hard	[CD79]
Trapezoids	polygons	N	$O(n^2)$	[Kei85]
Trapezoids	polygons	Y	NP-complete	[AAI86]
Rectangles	orthogonal	N	$O(n^{3/2} \log n)$	[ACKO88]
		Y	NP-complete	[ACKO88]

gon covers. An easy 4-approximation for minimum convex cover has been known for some time: shoot a ray from each reflex vertex, splitting into two convex angles [HM85]. Without Steiner points, balanced geometric separators have led to a quasi-PTAS [BBV15]. Building on [CR94], a polynomial-time approximation algorithm can achieve an approximation ratio of $O(\log n)$ [EW03].

STAR COVERS: ART GALLERY COVERAGE

Star-shaped covers of polygons corresponds to coverage of an "art gallery" by point-guards (Chapter 33). Vertex-guards is a further, useful restriction to place the guards at vertices. Because minimum star cover is APX-hard (Table 30.3.3) for both point- and vertex-guards [ESW01], the focus has been on approximation algorithms [Gho10]. An algorithm that runs in pseudopolynomial time achieves an approximation factor of $O(\log \mathrm{OPT})$ for point or perimeter guards [DKDS07]. The same approximation ratio was achieved via a randomized algorithm that runs in fully polynomial expected-time [EHP06]. Among the strongest results in this direction are an $O(\log \log \mathrm{OPT})$-approximation for vertex guards, running in $O(n^3)$, and an $O(\log h \log \mathrm{OPT})$-approximation that runs in $O(n^3h^2)$ for a polygon with h holes [Kin13].

FAT PARTITIONS

Because many algorithms work faster on "fat" shapes, partitioning polygons into fat pieces has become a recent focus. One notion of fatness asks for a partition into convex polygons that minimizes the largest aspect ratio of any piece of the partition. The ***aspect ratio*** of a polygon P is the ratio of the diameters of the smallest circumscribing circle to the largest inscribed circle. Thus, the fatness corresponds to circularity. If Steiner points are disallowed, i.e., if the pieces of the partition must have their vertices chosen among P's vertices, then a polynomial-time algorithm is known [DI04]. Permitting Steiner points leads to considerable complexity. For example, the optimal partition of an equilateral triangle needs an infinite number of pieces, and the optimal partition for a square is not yet known [DO03]. See Figure 30.3.1.

A variation on the Treemap algorithm, for a tree of n nodes and height h, can construct a convex partition into convex polygons with aspect ratio bounded by $O(\mathrm{poly}(h, \log n))$ [BOS13]. This then leads to a tree partition into fat rectangles with similarly bounded aspect ratio.

A chip manufacturing application has led to another type of "fat" rectangle partition, where the goal is to maximize the shortest rectangle side over all rect-

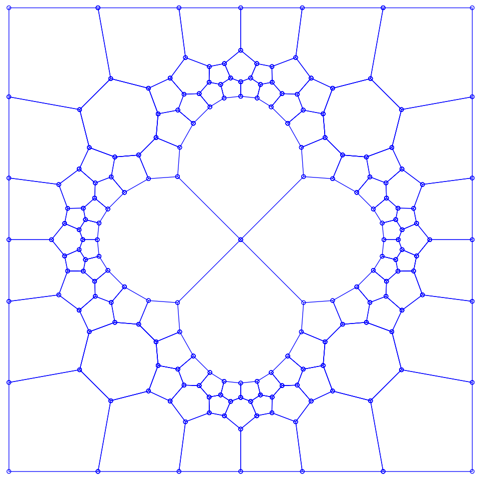

FIGURE 30.3.1
A 92-piece partition achieving an aspect ratio of 1.29950.

angles in the partition, i.e., avoid thin rectangles. The challenge is that the edges of the optimal partition need not be "anchored" to a point on the boundary of the polygon, but may instead float freely inside, as in Fig. 30.3.2. Nevertheless, for simple orthogonal polygons, a polynomial-time algorithm can be achieved, albeit with high complexity [OT04].

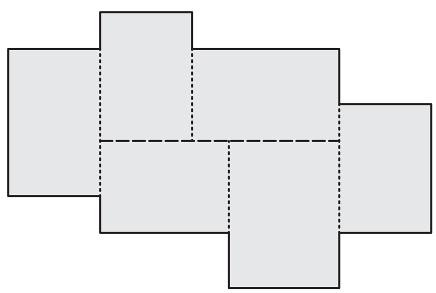

FIGURE 30.3.2
Not every cut of an optimal rectangle partition is "anchored" on the boundary.

ORTHOGONAL POLYGONS

Partitions and covers of orthogonal polygons into rectangles were mentioned above. With the goal of achieving the fewest number of rectangles, finding optimal covers is NP-complete, whereas finding optimal partitions is polynomial, $O(n^{3/2} \log n)$. If the goal is to minimize the total length of the "cuts" between the rectangles (minimum "ink"), then an optimum partition can be found in $O(n^4)$ time for polygons without holes, but is NP-complete with holes [LTL89]. Approximations are available; for example, one that guarantees a solution within a factor of 3 of the minimum length [GZ90]. Another variation on ink minimization

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry, J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

seeks a convex partition of a polygon P with the total perimeter of the convex pieces a factor of the perimeter of P. Allowing Steiner points, a total perimeter of $O((\log n/\log\log n)\mathrm{perim}(P))$ can be achieved [DT11]. Covering orthogonal polygons without holes with the fewest squares is polynomial, $O(n^{3/2})$, but NP-complete for polygons with holes [ACKO88].

AREA BISECTION

A particularly useful partition of a polygon P is an ***area bisection***: a line determining a halfplane H such that $H \cap P$ and $\bar{H} \cap P$ have the same area. In [DO90] an $O(n \log n)$ algorithm for area bisection was developed, and then used to "ham-sandwich section" a pair of polygons. This result was subsequently improved to $O(n)$-time [She92]. Motivated by positioning parts in industrial part-feeding systems, Böhringer et al. [BDH99] developed an output-size sensitive algorithm for computing the complete set of combinatorially distinct area bisectors, which they show can have size $\Omega(n^2)$.

A *ham-sandwich geodesic* in a polygon P enclosing points is a shortest path connecting two boundary points that simultaneously bisects red points and blue points in the polygon. If n is the number of vertices of P plus the number of interior points, such a geodesic can be found in $O(n \log r)$-time, where r is the number of reflex vertices of P [BDH$^+$07]. This result has been generalized to the situation where there are kn red points and km blue points, and the task is to partition P into k *relatively-convex regions* (closed under shortest paths), each containing n red and m blue points. Then a $O(kn^2 \log^2 n)$-time algorithm is available [BBK06]. A related result partitions a polygon of n vertices containing k points into equal-area convex regions, each containing exactly one point. $O(kn + k^2 \log n)$-time can be achieved [AP10].

A classic result known as Winternitz's theorem says that in every convex polygon P, there is a point $x \in P$ such that any halfplane that contains x contains at least 4/9's of P's area. This has been generalized to nonconvex polygons P of $r \geq 1$ reflex vertices: there is a point $x \in P$ such that any boundary-to-boundary segment chord partitions P into two pieces, such that the piece that contains x contains nearly a fraction $1/(2(r+1))$ of the area of P [BCHM11].

SUM-DIFFERENCE DECOMPOSITIONS

Permitting set subtraction as well as set union leads to natural shape decompositions. This is evident from the field of Constructive Solid Geometry, where shapes are described with CSG trees whose nodes are union or difference operators, and whose leaves are primitive shapes (Section 57.1). Batchelor developed a similar concept for shape description, the ***convex deficiency tree*** [Bat80]. For a shape P, the root of this tree is its hull $\mathrm{conv}(P)$, the children of the root the hulls of the convex deficiencies $\mathrm{conv}(P) \setminus P$, and so on [O'R98, p. 98].

Chazelle suggested [Cha79] representing a shape by the difference of convex sets: $A \setminus B$ where A and B are unions of convex polygons. It has been established that finding the minimum number of convex pieces in such a sum-difference decomposition of a multiply connected polygonal region is NP-hard [Con90].

DISSECTIONS

A ***dissection*** of one polygon P to another Q is a partition of P into a finite number of pieces that may be reassembled to form Q. P and Q are then said to be *equidecomposable*. Dissections have been studied as puzzles for centuries. A typical example is shown in Figure 30.3.3 [Fre97, p. 66]. It has been known since the early

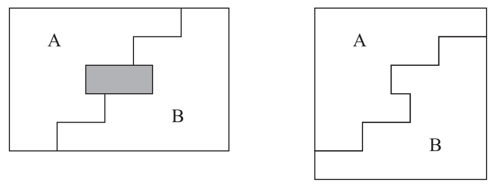

FIGURE 30.3.3
Sam Loyd's "A&P Baking Powder" puzzle reassembles a rectangle with a hole to a rectangle without a hole via a two-piece dissection.

19th century that any two polygons of equal area are equidecomposable [Fre97, p. 221]. The same question for the more constrained ***hinged dissections*** remained open (implicitly) for years. See Fig. 30.3.4 for the famous Dudeney-McElroy hinged dissection between a square and an equilateral triangle [Fre02]. Now the question is settled positively: any finite collection of polygons of equal area has a common hinged dissection [AAC$^+$12]. For two polygons with vertices on the integer lattice, both the number of pieces of the hinged dissection and the running time of the construction algorithm are pseudopolynomial.

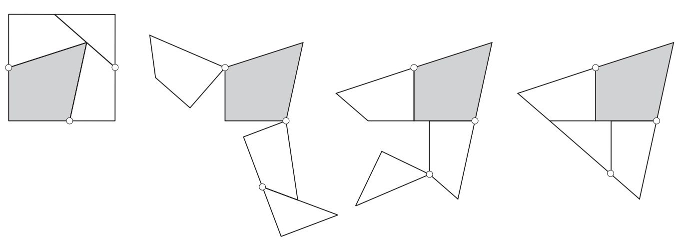

FIGURE 30.3.4
A four-piece hinged dissection between a square and an equilateral triangle.

OPEN PROBLEMS

1. *Approximating the number of art gallery guards:* Give a polynomial-time algorithm for computing a constant-factor approximation of the minimum number of point guards needed to cover a simple polygon.

2. *Fat partition of a square:* What is the optimal partition of a square into "fat" convex polygons?

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry, J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

30.4 POLYGON INTERSECTION

Polygon intersection problems deal with issues of detection and computation of the collision between two polygonal shapes. In the detection problem, one is only interested in deciding *whether* the two polygons have a point in common. In the intersection computation problem, the algorithm is asked to report the overlapping parts of the two polygons. Such problems arise naturally in robotics and computer games; see Chapter 39 for additional material.

The maximum *number* of points at which the boundaries of two polygons may cross each other depends on the type of polygons. If p and q, respectively, denote the number of vertices of the two polygons, then the maximum number of intersections is $\min(2p, 2q)$ if both polygons are convex, $\max(2p, 2q)$ if one is convex, and pq otherwise.

Algorithmically, intersection-detection between convex polygons can be done significantly faster than intersection computation, if we allow reasonable preprocessing of polygons. By a reasonable preprocessing, we mean that the preprocessing algorithm takes into account the *structure* of the polygons but *not their positions.* In Table 30.4.1, n denotes the total number of vertices in the two polygons; that is, $n = p + q$.

TABLE 30.4.1 Intersecting polygons.

POLYGON TYPES	PREPROCESSING	QUERY	SOURCE
Convex-convex	$O(1)$	$O(n)$	[CD80]
Convex-convex	$O(n)$	$O(\log n)$	[CD80]
Simple-simple	$O(1)$	$O(n)$	[Cha91]
Simple-simple	$O(n \log n)$	$O(m \log^2 n)$	[Mou92]

The parameter m in the query time for intersections of two simple polygons is the complexity of a *minimum link witness* for the intersection or disjointness of the two polygons, and we always have $m \leq n$. The preprocessing space requirement is linear when the polygons are preprocessed.

30.5 POLYGON CONTAINMENT

Polygon containment refers to a class of problems that deals with the placement of one polygonal figure inside another. Polygon inscription, polygon circumscription, and polygon nesting are other variants of this type of problem.

GLOSSARY

Inscribed polygon: We will say that a polygon Q is inscribed in polygon P if $Q \subset P$. P is then called a ***circumscribing polygon***.

Polygon nesting: P, Q is a nested pair if $Q \subset P$ or vice versa.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry, J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

CONTAINMENT OF POLYGONS

Let P, Q be two simple polygons with p and q vertices, respectively. The polygon containment problem asks for the largest copy of Q that can be contained in P using rotations, translations, and scaling. (In this section, all scalings are assumed to be *uniform*; thus "shearing" is not permitted.) The containment problems can be solved using the parameter space of all translated or rotated copies of P, by computing the *free space* of placements that lie in Q; see Section 50.2 for further details). The largest scale factor of P for which such a placement exists can be found with parametric search. Table 30.5.1 collects the best results known for the most important cases. See Section 28.10 for a description of the near-linear λ_s function.

TABLE 30.5.1 Results for the polygon containment problem.

P	Q	TRANSFORMS	RESULTS	SOURCE
Convex	convex	translate, scale	$O(p + q\log q)$	[ST94, GP13]
Convex	polygon w holes	translate, scale	$O(pq\log(pq))$	[For85, LS87]
Orthogonal	orthogonal	translate, scale	$O(pq\log(pq))$	[Bar96]
Simple	polygon w holes	translate, scale	$O(p^2q^2\log(pq))$	[AFH02]
Convex	convex	translate, rotate, scale	$O(pq^2\log q)$	[AAS98]
Convex	polygon w holes	translate, rotate, scale	$O(pq\lambda_6(pq)\log^c(pq))$	[AAS99]
Simple	polygon w holes	translate, rotate, scale	$O(p^3q^3\log pq)$	[AB88]

It has been shown recently that the decision problem—whether there exists a transformation of Q that permits it to be contained in P—is 3SUM-hard for simple polygons under homotheties and for convex polygons under similarities [BHP01]. Despite recent breakthroughs on the computational complexity of 3SUM [Fre15], it is unlikely the above bounds can be pushed much below quadratic [AVWY15].

When only rigid motions are allowed and P does not contain any copy of Q, one can seek to maximize the area of the intersection of P and a copy of Q. Under translations, this can be done in $O((p+q)\log(pq))$ time for two convex polygons [BCD$^+$98], and in $O(p^2q^2)$ time for two simple polygons [MSW96]; an $(1-\varepsilon)$-approximation is available in $O(p+q)$ time for every $\varepsilon > 0$ [HPR14]. Under translations and rotations, only approximation algorithms are known [CL13, HPR14].

INSCRIBING/CIRCUMSCRIBING POLYGONS

We now consider problems related to inscribing and circumscribing polygons. In these problems, a polygon P is given, and the task is to find a polygon Q of some specified number of vertices k that is inscribed in (resp. circumscribes) P while maximizing (resp. minimizing) certain measures of Q. The common measures include area and perimeter. See Table 30.5.2 for results concerning this class of problems; n denotes the number of vertices of P.

TABLE 30.5.2 Inscribing and circumscribing polygons.

TYPE	k	P	MEASURE	RESULTS	SOURCE
Inscribe	3	convex	max area	$O(n \log n)$	[KLU+17]
Inscribe	k	convex	max area/perimeter	$O(kn + n \log n)$	[AKM+87, KLU+17]
Inscribe	convex	simple	max area, perimeter	$O(n^7)$, $O(n^6)$	[CY86]
Inscribe	3	simple	max area/perimeter	$O(n^4)$	[MS90]
Circumscribe	3	convex	min area	$O(n)$	[OAMB86]
Circumscribe	3	convex	min perimeter	$O(n)$	[BM02]
Circumscribe	k	convex	min area	$O(kn + n \log n)$	[AP88]

NESTING POLYGONS

The nested polygon problem asks for a polygon with the smallest number of vertices that fits between two nested polygons. More precisely, given two nested polygons P and Q, where $Q \subset P$, find a polygon K of the least number of vertices such that $Q \subset K \subset P$. Generalizing the notion of nested polygons, one can also pose the problem of determining a polygonal subdivision of the least number of *edges* that "separates" a family of polygons. Table 30.5.3 lists the results on these problems. In this table, n is the total number of vertices in the input polygons, while k is the number of vertices in the output polygon (or subdivision).

TABLE 30.5.3 Results for polygon nesting.

TYPES OF P, Q	TYPE OF K	RESULTS	SOURCE
Convex-convex	convex	$O(n \log k)$	[ABO+89]
Simple-simple	simple	$O(n \log k)$	[Gho91]
Polygonal family	subdivision	NP-complete	[Das90]
Polygonal family	subdivision	$O(1)$-Opt in $O(n \log n)$	[MS95b]

Several other results on polygon nesting have been obtained. In particular, if the minimum-vertex nested polygon is nonconvex, then it can be found in $O(n)$ time [GM90]. There is also a relation here to *offset polygons* [BG14] (Chapter 57), and *minimum-link separators* (Chapter 53).

OPEN PROBLEMS

1. *Large empty convex polygons:* Danzer conjectured that for every set S of n points in the unit square $[0,1]^2$, there is a convex polygon in $[0,1]^2 \setminus S$ of area $\Omega(1/\log n)$. There are $(1-\varepsilon)$-approximation algorithms [DHPT14] for finding the maximum area of a convex polygon in $[0,1]^2 \setminus S$.

2. *Square Peg Problem:* Toeplitz [Toe11] conjectured that every Jordan curve C in the plane contains four points that are the vertices of a square. The

conjecture has been confirmed in many important special cases, such as piecewise linear curves (e.g., the boundaries of polygons) [Pak10] or smooth curves [CDM14], but remains open in general; see [Mat14] for a survey.

30.6 MISCELLANEOUS

There is a rather large number of results pertaining to polygons, and it would be impossible to cover them all in a single chapter. Having focused on a selected list of topics so far, we now provide below an unorganized collection of some miscellaneous results.

POLYGON MORPHING

To ***morph*** one polygon into another is to find a continuous deformation from the source polygon to the target polygon. In a *parallel morph*, the deformation maintains the orientation of every edge of the polygon. A parallel morph exists between any two simple n-gons if their edges, taken in counterclockwise order, are parallel and oriented the same way [GCK91]. Hershberger and Suri [GHS00] show that a sequence of $O(n \log n)$ morphing steps suffice where each step consists of a uniform scaling or translation of a part of the polygon.

Between two arbitrary simple n-gons in the plane, there is always a morph in $O(n)$ steps such that in each step all vertices move at constant speed along parallel lines [AAB$^+$17]. Such a morph exists, in general, between any two isotopic straight-line embeddings of a planar graph G; the case of two simple polygons with n vertices corresponds to $G = C_n$. The morph can be computed in $O(n^3)$ time; it crucially relies on the case of morphing triangulations. Aronov et al. [ASS93] showed that for any two simple n-gons, P and Q, there exists straight-line isotopic triangulations (so-called *compatible triangulations*) using $O(n^2)$ Steiner points, and this bound cannot be improved.

Morphing between simple polygons of fixed edge lengths (i.e., linkages) is discussed in Chapter 9.

CSG REPRESENTATION

Peterson proved that every simple polygon in two dimensions admits a representation by a Boolean formula on the halfplanes supporting the edges of the polygon. Furthermore, the resulting formula is *monotone*; that is, there is no negation and each halfplane appears exactly once. A ***Peterson-style formula*** is a "constructive solid geometry" representation, in which the polygon is presented as a set of Boolean operations; see Chapter 57. Interestingly, it turns out that not all 3D polyhedra admit a Peterson-style formula [DGHS93].

Dobkin et al. [DGHS93] give an $O(n \log n)$ time algorithm for computing a Petersen-style formula for a simple n-gon. Chirst et al. [CHOU10] show that every n-gon can be expressed as a monotone Boolean formula of at most $\lfloor (4n2)/5 \rfloor$ wedges (where a wedge is the intersection or the union of two halfplanes), and $\lceil (3n4)/5 \rceil$ wedges are sometimes necessary.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry,
J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

POLYGON SEARCHING AND PURSUIT-EVASION

In these problems, the goal is to design search strategies for an (identifiable) object (intruder) in a polygon. The motivation often comes from surveillance applications in robotics. The "polygon searching" line of research typically assumes that the object of search is *stationary*, the searcher "discovers" the geometry of the polygon during its navigation (*on-line* model), and the goal is to minimize the search cost (distance traveled), measured by its competitive ratio. Table 30.6.1 summarizes some basic results on the polygon searching problems. (The parameter k in the second to last line denotes the number of distinct initial placements of the robot having the same visibility polygon.) The survey article [GK10] is a good starting point for this topic.

TABLE 30.6.1 Results for polygon searching.

ENVIRONMENT	GOAL	COMPETITIVE RATIO	SOURCE
n oriented rectangles	shortest path	$\Theta(\sqrt{n})$	[BRS97]
"Street" polygon	shortest path	$\sqrt{2}$	[IKLS04]
Star-shaped polygon	reach kernel	≈ 3.12	[Pal00]
Orthogonal polygon	exploration	randomized 5/4	[Kle94]
Simple polygon	localization with min travel	$(k-1)$-Opt	[DRW98]
Simple polygon	shortest watchman tour	26.5-Opt	[HIKK01]

In the "pursuit-evasion" line of research, one or more searchers (pursuers) coordinate to locate and capture a *mobile* object (intruder), and the goal is to establish necessary and sufficient conditions for a successful pursuit. The survey article [CHI11] is a good starting point for this topic. The origin of pursuit-evasion goes back to the celebrated "Lion-and-Man" problem, attributed to Rado in 1930s: if a man and a lion are confined to a closed arena, and both have equal maximum speeds, can the lion catch the man? Surprisingly, the man can evade the lion indefinitely as shown by Besicovitch [Lit86]—the lion fails to reach the man in any finite time although it can get arbitrarily close to him. In computational geometry, a primary focus of research is to bound the minimum number of pursuers needed to locate or capture the evader, as a function of environment's complexity. The model assumes that pursuers and the evader can move with the same maximum speed, the geometry of the environment (polygon) is known to all players, and the players move taking alternating turns. Two models of visibility are considered: in the *full visibility* model, each player knows the position of all other players at all times (following the convention of the cops-and-robber game in graphs) while in the *LoS visibility* model, each player is limited to its line-of-sight visibility. Table 30.6.2 below summarizes the current state of the art. (The *minimum feature size* (MFS) condition requires that the minimum (geodesic) distance between two vertices is lower bounded by the distance each player can move in one step.) Open problems include closing the gap between upper and lower bounds as well as extending the pursuit to three-dimensional environments.

TABLE 30.6.2 Results for pursuit-evasion in polygons.

ENVIRONMENT	GOAL	VISIBILITY MODEL	NUMBER OF PURSUERS	SOURCE
Simple polygon	Locate	LoS	$\Theta(\log n)$	[GLL$^+$99]
Polygon with h holes	Locate	LoS	$\Theta(\log n + \sqrt{h})$	[GLL$^+$99]
Polygon with h holes	Capture	Full	3	[BKIS12]
Polygon with holes	Capture	LoS without MFS	$\Omega(n^{2/3})$, $O(n^{5/6})$	[KS15a]
Polygon with h holes	Capture	LoS with MFS	$O(\log n + \sqrt{h})$	[KS15a]
3D polyhedral surface	Capture	Full	3 nec., 4 suff.	[KS15b]

THREE-DIMENSIONAL POLYGONS

A 3D polygon is an unknotted closed chain of segments in $\mathbb{R}^3$ such that adjacent segments share an endpoint, and nonadjacent segments do not intersect. A triangulation of a 3D polygon has the same combinatorial structure as a triangulation of a planar polygon—all triangle vertices are polygon vertices, each polygon edge is a side of one triangle, each diagonal is shared by exactly two triangles—with the surface they define a nonself-intersecting topological disk. This disk is said to *span* the polygon. Barequet et al. [BDE98] proved that determining whether a 3D polygon has a triangulation in this sense is NP-complete. Another negative result along the same lines is that there exist 3D polygons of n vertices that can only be spanned by nonself-intersecting piecewise-linear disks which, when triangulated, need $2^{\Omega(n)}$ triangles [HST03]. Note that here the triangle vertices are not necessarily polygon vertices, i.e., Steiner points are (necessarily) used. This exponential lower bound shows that ***knot triviality*** algorithms (which check whether a closed chain is the trivial "unknot") that search for such spanning disks necessarily lead to exponential-time algorithms [Bur04, Lac15]. This unknotting problem is known to be in NP [HLP99] and co-NP [Lac16].

OPEN PROBLEMS

1. *Random Polygonizations.* Is there an efficient algorithm to generate a random polygonization for n given points in the plane?
2. *3D Peterson formulas:* Characterize the 3D polyhedra that can be represented by Peterson-style formulas.
3. *Computational complexity of unknot recognition:* Is there a polynomial-time algorithm for deciding whether a 3D polygon is a trivial knot.

30.7 SOURCES AND RELATED MATERIAL

SURVEYS

The survey article by Mitchell and Suri [MS95a] addresses optimization problems in computational geometry, many involving polygons. Keil surveys polygon decomposition algorithms in [Kei00]. Link distance problems are surveyed in [MSD00].

RELATED CHAPTERS

Chapter 29: Triangulations and mesh generation
Chapter 31: Shortest paths and networks
Chapter 33: Visibility
Chapter 38: Point location
Chapter 54: Pattern recognition
Chapter 59: Geographic information systems

REFERENCES

[AAB+17] S. Alamdari, P. Angelini, F. Barrera-Cruz, T.M. Chan, G. Da Lozzo, G. Di Battista, F. Frati, P. Haxell, A. Lubiw, M. Patrignani, V. Roselli, S. Singla, and B.T. Wilkinson. How to morph planar graph drawings. *SIAM J. Comput.*, 46:824–852, 2017.

[AAC+12] T.G. Abbott, Z. Abel, D. Charlton, E.D. Demaine, M.L. Demaine, and S.D. Kominers. Hinged dissections exist. *Discrete Comput. Geom.*, 47:150–186, 2012.

[AAI86] Ta. Asano, Te. Asano, and H. Imai. Partitioning a polygonal region into trapezoids. *J. ACM*, 33:290–312, 1986.

[AAK09] E. Ackerman, O. Aichholzer, and B. Keszegh. Improved upper bounds on the reflexivity of point sets. *Comput. Geom.*, 42:241–249, 2009.

[AAS98] P.K. Agarwal, N. Amenta, and M. Sharir. Largest placement of one convex polygon inside another. *Discrete Comput. Geom.* 19:95–104, 1998.

[AAS99] P.K. Agarwal, B. Aronov, and M. Sharir. Motion planning for a convex polygon in a polygonal environment. *Discrete Comput. Geom.*, 22:201–221, 1999.

[AB88] F. Avnaim and J.-D. Boissonnat. Polygon placement under translation and rotation. *Proc. 5th Sympos. Theoret. Aspects Comput. Sci.*, vol. 294 of *LNCS*, pages 322–333, Springer, Berlin, 1988.

[ABO+89] A. Aggarwal, H. Booth, J. O'Rourke, S. Suri, and C.K. Yap. Finding minimal convex nested polygons. *Inform. Comput.*, 83:98–110, 1989.

[ACKO88] L.J. Aupperle, H.E. Conn, J.M. Keil, and J. O'Rourke. Covering orthogonal polygons with squares. In *Proc. 26th Allerton Conf. Commun. Control Comput.*, pages 97–106, 1988.

[AFH02] P.K. Agarwal, E. Flato, and D. Halperin. Polygon decomposition for efficient construction of Minkowski sums. *Comput. Geom.*, 21:3961, 2002.

[AFH+03] E.M. Arkin, S.P. Fekete, F. Hurtado, J.S.B. Mitchell, M. Noy, V. Sacristán, and S. Sethia. On the reflexivity of point sets. In *Discrete Comput. Geom.*, pages 139–156, 2003.

[Agg84] A. Aggarwal. *The art gallery problem: Its variations, applications, and algorithmic aspects.* Ph.D. thesis, Dept. Comput. Sci., Johns Hopkins Univ., Baltimore, 1984.

[AGR01] N.M. Amato, M.T. Goodrich, and E.A. Ramos. A randomized algorithm for triangulating a simple polygon in linear time. *Discrete Comput. Geom.*, 26:245–265, 2001.

[AHST03] O. Aichholzer, M. Hoffmann, B. Speckmann, and C.D. Tóth. Degree bounds for constrained pseudo-triangulations. In *Canad. Conf. Comput. Geom.*, pages 155–158, 2003.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry, J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

[AKM⁺87] A. Aggarwal, M.M. Klawe, S. Moran, P.W. Shor, and R. Wilber. Geometric applications of a matrix-searching algorithm. *Algorithmica*, 2:195–208, 1987.

[AP88] A. Aggarwal and J.K. Park. Notes on searching in multidimensional monotone arrays. In *Proc. 29th IEEE Sympos. Found. Comput. Sci.*, pages 497–512, 1988.

[AP10] D. Adjiashvili and D. Peleg. Equal-area locus-based convex polygon decomposition. *Theoret. Comput. Sci.*, 411:1648–1667, 2010

[ASS93] B. Aronov, R. Seidel, and D.L. Souvaine. On compatible triangulations of simple polygons. *Comput. Geom.*, 3:27–35, 1993.

[AVWY15] A. Abboud, V. Vassilevska Williams, and H. Yu. Matching triangles and basing hardness on an extremely popular conjecture. In *Proc. 47th ACM Sympos. Theory Comput.*, pages 41–50, 2015.

[Bar96] A.H. Barrera. Algorithms for deciding the containment of polygons. *Inform. Process. Lett.*, 59:261–265, 2996.

[Bat80] B.G. Batchelor. Hierarchical shape description based upon convex hulls of concavities. *J. Cybern.*, 10:205–210, 1980.

[BBK06] S. Bereg, P. Bose, and D. Kirkpatrick. Equitable subdivisions within polygonal regions. *Comput. Geom.*, 34:20–27, 2006.

[BBV15] S. Bandyapadhyay, S. Bhowmick, and K. Varadarajan. Approximation schemes for partitioning: Convex decomposition and surface approximation. In *Proc. 26th ACM-SIAM Symp. Discrete Algorithms*, pages 1457–1470, 2015.

[BCD⁺98] M. de Berg, O. Cheong, O. Devillers, M. van Kreveld, and M. Teillaud. Computing the maximum overlap of two convex polygons under translations. *Theory Comput. Syst.*, 31:613–628, 1998.

[BCHM11] P. Bose, P. Carmi, F. Hurtado, and P. Morin. A generalized Winternitz theorem. *J. Geometry*, 100:29–35, 2011.

[BDE98] G. Barequet, M.T. Dickerson, and D. Eppstein. On triangulating three-dimensional polygons. *Comput. Geom.*, 10:155–170, 1998.

[BDH99] K.-F. Böhringer, B.R. Donald, and D. Halperin. The area bisectors of a polygon. *Discrete Comput. Geom.*, 22:269–285, 1999.

[BDH⁺07] P. Bose, E.D. Demaine, F. Hurtado, J. Iacono, S. Langerman, and P. Morin. Geodesic ham-sandwich cuts. *Discrete Comput. Geom.*, 37:325–339, 2007.

[BG14] G. Barequet and A. Goryachev. Offset polygon and annulus placement problems. *Comput. Geom.*, 47:407–434, 2014.

[BHP01] G. Barequet and S. Har-Peled. Polygon containment and translational min-Hausdorff-distance between segment sets are 3SUM-hard. *Internat. J. Comput. Geom. Appl.*, 11:465–474, 2001.

[BKIS12] D. Bhadauria, K. Klein, V. Isler, and S. Suri. Capturing an evader in polygonal environments with obstacles: The full visibility case. *I. J. Robot. Res.*, 31:1176–1189, 2012.

[BM02] B. Bhattacharya and A. Mukhopadhyay. On the minimum perimeter triangle enclosing a convex polygon. In J. Akiyama and M. Kano, editors, *Discrete and Computational Geometry*, vol. 2866 of *LNCS*, pages 84–96, Springer, Berlin, 2002.

[BOS13] M. de Berg, K. Onak, and A. Sidiropoulos. Fat polygonal partitions with applications to visualization and embeddings. *J. Comput. Geom.*, 4:212–239, 2013.

[BRS97] A. Blum, P. Raghavan, and B. Schieber. Navigating in unfamiliar geometric terrain. *SIAM J. Comput.*, 26:110–137, 1997.

[Bur04] B.A. Burton. Introducing Regina, the 3-manifold topology software. *Exp. Math.*, 13:267–272, 2004.

[CD79] B. Chazelle and D.P. Dobkin. Decomposing a polygon into its convex parts. In *Proc. 11th ACM Sympos. Theory Comput.*, pages 38–48, 1979.

[CD80] B. Chazelle and D.P. Dobkin. Detection is easier than computation. In *Proc. 12th ACM Sympos. Theory Comput.*, pages 146–153, 1980.

[CDM14] J. Cantarella, E. Denne, and J. McCleary. Transversality for configuration spaces and the "Square-Peg" Theorem. Preprint, `arXiv:1402.6174`, 2014.

[Cha79] B. Chazelle. *Computational geometry and convexity.* Ph.D. thesis, Dept. Comput. Sci., Yale Univ., New Haven, 1979. Carnegie-Mellon Univ. Report CS-80-150.

[Cha82] B. Chazelle. A theorem on polygon cutting with applications. In *Proc. 23rd IEEE Sympos. Found. Comput. Sci.*, pages 339–349, 1982.

[Cha91] B. Chazelle. Triangulating a simple polygon in linear time. *Discrete Comput. Geom.*, 6:485–524, 1991.

[CHI11] T.H. Chung, G.A. Hollinger, and V. Isler. Search and pursuit-evasion in mobile robotics. *Autonomous Robots*, 31:299–316, 2011.

[CHOU10] T. Christ, M. Hoffmann, Y. Okamoto, and T. Uno. Improved bounds for wireless localization. *Algorithmica*, 57:499–516, 2010.

[Chr11] T. Christ. Beyond triangulation: covering polygons with triangles. In *Proc. 12th Workshop Algorithms Data Structures*, vol. 6844 of *LNCS*, pages 231–242, Springer, Berlin, 2011.

[CL13] S.-W. Cheng and C.-K. Lam. Shape matching under rigid motion. *Comput. Geom.*, 46:591–603, 2013.

[Con90] H. Conn. *Some Polygon Decomposition Problems.* Ph.D. thesis, Dept. Comput. Sci., John Hopkins Univ., Baltimore, 1990.

[CR94] J.C. Culberson and R.A. Reckhow. Covering polygons is hard. *J. Algorithms*, 17:2–24, 1994.

[CY86] J.S. Chang and C.K. Yap. A polynomial solution for the potato-peeling problem. *Discrete Comput. Geom.*, 1:155–182, 1986.

[Das90] G. Das. *Approximation schemes in computational geometry.* Ph.D. thesis, Univ. of Wisconsin, 1990.

[DFH+99] L. Devroye, P. Flajolet, F. Hurtado, M. Noy, and W. Steiger. Properties of random triangulations and trees. *Discrete Comput. Geom.*, 22:105–117, 1999.

[DFOR10] M. Damian, R. Flatland, J. ORourke, and S. Ramaswami. Connecting polygonizations via stretches and twangs. *Theory Comput. Syst.*, 47:674–695, 2010.

[DGHS93] D.P. Dobkin, L.J. Guibas, J. Hershberger, and J. Snoeyink. An efficient algorithm for finding the CSG representation of a simple polygon. *Algorithmica*, 10:1–23, 1993.

[DHPT14] A. Dumitrescu, S. Har-Peled, and C.D. Tóth. Minimum convex partitions and maximum empty polytopes. *J. Comput. Geom.*, 5:86–103, 2014.

[DI04] M. Damian. Exact and approximation algorithms for computing optimal fat decompositions. *Comput. Geom.*, 28:19–27, 2004.

[DKDS07] A. Deshpande, T. Kim, E.D. Demaine, and S.E. Sarma. A pseudopolynomial time $o(\log n)$-approximation algorithm for art gallery problems. In *Proc. 10th Workshop Algorithms Data Structs.*, vol. 4619 of *LNCS*, pages 163–174, Springer, Berlin, 2007.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry,
J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

[DO90] M. Díaz and J. O'Rourke. Ham-sandwich sectioning of polygons. In *Proc. 2nd Canad. Conf. Comput. Geom.*, pages 282–286, 1990.

[DO03] M. Damian and J. O'Rourke. Partitioning regular polygons into circular pieces I: Convex partitions. In *Proc. 15th Canad. Conf. Comput. Geom.*, pages 43–46, 2003.

[DRW98] G. Dudek, K. Romanik, and S. Whitesides. Localizing a robot with minimum travel. *SIAM J. Comput.*, 27:583–604, 1998.

[DQTW05] Q. Ding, J. Qian, W. Tsang, and C. Wang. Randomly generating triangulations of a simple polygon. In *Proc. 11th Computing Combin. Conf.*, vol. 3595 of *LNCS*, pages 471–480, Springer, Berlin, 2005.

[DT10] A. Dumitrescu and C.D. Tóth. Long non-crossing configurations in the plane. *Discrete Comput. Geom.*, 44:727–752, 2010.

[DT11] A. Dumitrescu and C.D. Tóth. Minimum weight convex Steiner partitions. *Algorithmica*, 60:627–652, 2011.

[EHP06] A. Efrat and S. Har-Peled. Guarding galleries and terrains. *Inform. Process. Lett.*, 100:238–245, 2006.

[ES94] P. Epstein and J.-R. Sack. Generating triangulations at random. *ACM Trans. Model. Comput. Simulation*, 4:267–278, 1994.

[ESW01] S.J. Eidenbenz, C. Stamm, and P. Widmayer. Inapproximability results for guarding polygons and terrains. *Algorithmica*, 31:79–113, 2001.

[EW03] S.J. Eidenbenz and P. Widmayer. An approximation algorithm for minimum convex cover with logarithmic performance guarantee. *SIAM J. Comput.*, 32:654–670, 2003.

[Fek00] S.P. Fekete. On simple polygonalizations with optimal area. *Discrete Comput. Geom.*, 23:73–110, 2000.

[FKMU13] R. Fulek, B. Keszegh, F. Morić, and I. Uljarević. On polygons excluding point sets. *Graphs Combin.*, 29:1741–1753, 2013.

[For85] S.J. Fortune. A fast algorithm for polygon containment by translation. In *Proc. 12th Internat. Colloq. Automata Lang. Program.*, vol. 194 of *LNCS*, pages 189–198, Springer, Berlin, 1985.

[Fre97] G.N. Frederickson. *Dissections: Plane and Fancy.* Cambridge University Press, 1997.

[Fre02] G.N. Frederickson. *Hinged Dissections: Swinging & Twisting.* Cambridge University Press, 2002.

[Fre15] A. Freund. Improved subquadratic 3SUM. *Algorithmica*, online first, 2015.

[GCK91] U. Grenander, Y. Chow, and D.M. Keenan. *Hands: A Pattern Theoretic Study of Biological Shapes.* Springer-Verlag, Berlin, 1991.

[Gho91] S.K. Ghosh. Computing visibility polygon from a convex set and related problems. *J. Algorithms*, 12:75–95, 1991.

[Gho10] S.K. Ghosh. Approximation algorithms for art gallery problems in polygons. *Discrete Applied Math.*, 158:718–722, 2010.

[GHS00] L. Guibas, J. Hershberger, and S. Suri. Morphing simple polygons. *Discrete Comput. Geom.*, 24:1–3, 2000.

[GJPT78] M.R. Garey, D.S. Johnson, F.P. Preparata, and R.E. Tarjan. Triangulating a simple polygon. *Inform. Process. Lett.*, 7:175–179, 1978.

[GK10] S.K. Ghosh, and R. Klein. Survey: online algorithms for searching and exploration in the plane. *Computer Science Review*, 4:189–201, 2010.

[GLL+99] L.J. Guibas, J.-C. Latombe, S.M. LaValle, D. Lin, and R. Motwani. Visibility-based pursuit-evasion in a polygonal environment. *Internat. J. Comput. Geom. Appl.*, 9:471–494, 1999.

[GM90] S.K. Ghosh and A. Maheshwari. An optimal algorithm for computing a minimum nested nonconvex polygon. *Inform. Process. Lett.*, 36:277–280, 1990.

[GNT00] A. Garcı, M. Noy, and J. Tejel. Lower bounds on the number of crossing-free subgraphs of K_n. *Comput. Geom.*, 16:211–221, 2000.

[GP13] M.T. Goodrich and P. Pszona. Cole's parametric search technique made practical. In *Proc. 25th Canad. Conf. Comput. Geom.*, Waterloo, ON, 2013.

[GZ90] T.F. Gonzalez and S.-Q. Zheng. Approximation algorithms for partitioning a rectangle with interior points. *Algorithmica*, 5:11–42, 1990.

[HIKK01] F. Hoffmann, C. Icking, R. Klein, and K. Kriegel. The polygon exploration problem. *SIAM J. Comput.*, 31:577–600, 2001.

[HLP99] J. Hass, J.C. Lagarias, and N. Pippenger. The computational complexity of knot and link problems. *J. ACM*, 46:185–211, 1999.

[HM85] S. Hertel and K. Mehlhorn. Fast triangulation of the plane with respect to simple polygons. *Inform. Control*, 64:52–76, 1985.

[HMO+09] F. Hurtado, C. Merino, D. Oliveros, T. Sakai, J. Urrutia, and I. Ventura. On polygons enclosing point sets II. *Graphs Combin.*, 25:327–339, 2009.

[HPR14] S. Har-Peled and S. Roy. Approximating the maximum overlap of polygons under translation. *Algorithmica*, online first, 2016.

[HS95] J. Hershberger and S. Suri. A pedestrian approach to ray shooting: Shoot a ray, take a walk. *J. Algorithms*, 18:403–431, 1995.

[HST03] J. Hass, J. Snoeyink, and W.P. Thurston. The size of spanning disks for polygonal curves. *Discrete Comput. Geom.*, 29:1–18, 2003.

[IKLS04] C. Icking, R. Klein, E. Langetepe, S. Schuierer, and I. Semrau. An optimal competitive strategy for walking in streets. *SIAM J. Comput.*, 33:462–486, 2004.

[IM11] J. Iwerks and J.S.B. Mitchell. Spiral serpentine polygonization of a planar point set. In *Proc. XIV Spanish Meeting Comput. Geom.*, vol. 7579 of *LNCS*, pages 146–154, Springer, Berlin, 2011.

[Kei85] J.M. Keil. Decomposing a polygon into simpler components. *SIAM J. Comput.*, 14:799–817, 1985.

[Kei00] J.M. Keil. Polygon decomposition. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 491–518, Elsevier, Amsterdam, 2000.

[Kin13] J. King. Fast vertex guarding for polygons with and without holes. *Comput. Geom.*, 46:219–231, 2013.

[Kle94] J. Kleinberg. On-line search in a simple polygon. In *Proc. 5th ACM-SIAM Sympos. Discrete Algorithms*, pages 8–15, 1994.

[Kli80] G.T. Klincsek. Minimal triangulations of polygonal domains. *Discrete Math.*, 9:121–123, 1980.

[KLU+17] V. Keikha, M. Löffler, J. Urhausen, I. van der Hoog. Maximum-area triangle in a convex polygon, revisited. Preprint, `arXiv:1705.11035`, 2017.

[KS02] J.M. Keil and J. Snoeyink. On the time bound for convex decomposition of simple polygons. *Int. J. Comput. Geom. Appl.*, 12,181–192, 2002.

[KS15a] K. Klein and S. Suri. Capture bounds for visibility-based pursuit evasion. *Comput. Geom.*, 48:205–220, 2015.

[KS15b] K. Klein and S. Suri. Pursuit evasion on polyhedral surfaces. *Algorithmica*, 73:730–747, 2015.

[Lac15] M. Lackenby. A polynomial upper bound on Reidemeister moves. *Ann. Math. (2)*, 182:491–564, 2015.

[Lac16] M. Lackenby. The efficient certification of knottedness and Thurston norm. Preprint, `arXiv:1604.00290`, 2016.

[Lit86] J.E. Littlewood. *Littlewood's Miscellany.* Cambridge University Press, 1986.

[LS87] D. Leven and M. Sharir. Planning a purely translational motion for a convex object in two-dimensional space using generalized Voronoi diagrams. *Discrete Comput. Geom.*, 2:9–31, 1987.

[LTL89] W.T. Liou, J.J.M. Tan, and R.C.T. Lee. Minimum partitioning simple rectilinear polygons in $O(n \log\log n)$ time. In *Proc. 5th Sympos. Comput. Geom.*, pages 344–353, ACM Press, 1989.

[Mas78] W.J. Masek. Some NP-complete set covering problems. Manuscript, MIT, Cambridge, 1978.

[Mat14] B. Matschke. A survey on the square peg problem. *Notices Amer. Math. Soc.*, 61:346–352, 2014.

[Mou92] D.M. Mount. Intersection detection and separators for simple polygons. In *Proc. 8th Sympos. Comput. Geom.*, pages 303–311, ACM Pres, 1992.

[MS90] E.A. Melissaratos and D.L. Souvaine. On solving geometric optimization problems using shortest paths. In *Proc. 6th Sympos. Comput. Geom.*, pages 350–359, ACM Press, 1990.

[MS95a] J.S.B. Mitchell, and S. Suri. Geometric algorithms. In M.O. Ball, T.L. Magnati, C.L. Monma, and G.L. Nemhauser, editors, *Handbook of Operations Research/Management Science*, pages 425–479, Elsevier, Amsterdam, 1995.

[MS95b] J.S.B. Mitchell and S. Suri. Separation and approximation of polyhedral surfaces. *Comput. Geom.*, 5:95–114, 1995.

[MSD00] A. Maheshwari, J.-R. Sack, and H.N. Djidjev. Link distance problems. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 519–558, Elsevier, Amsterdam, 2000.

[MSW96] D.M. Mount, R. Silverman, and A.Y. Wu. On the area of overlap of translated polygons. *Comp. Vis. Image Understanding* 64:5361, 1996.

[OAMB86] J. O'Rourke, A. Aggarwal, S. Maddila, and M. Baldwin. An optimal algorithm for finding minimal enclosing triangles. *J. Algorithms*, 7:258–269, 1986.

[O'R87] J. O'Rourke. *Art Gallery Theorems and Algorithms. Internat. Ser. Monogr. Comput. Sci.*, Oxford University Press, New York, 1987.

[O'R98] J. O'Rourke. *Computational Geometry in C*, second edition. Cambridge University Press, 1998.

[OS83] J. O'Rourke and K.J. Supowit. Some NP-hard polygon decomposition problems. *IEEE Trans. Inform. Theory*, IT-30:181–190, 1983.

[OT04] J. O'Rourke and G. Tewari. The structure of optimal partitions of orthogonal polygons into fat rectangles. *Comput. Geom.*, 28(1):49–71, 2004.

[Pak10] I. Pak. *Lectures on Discrete and Polyhedral Geometry*, Manuscript, `http://www.math.ucla.edu/~pak/book.htm`, 2010.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry,
J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.

[Pal00] L. Palios. A new competitive strategy for reaching the kernel of an unknown polygon. *Proc. 7th Scandinavian Workshop Algorithm Theory*, vol. 1851 of *LNCS*, pages 367–382, Springer, Berlin, 2000.

[Ror14] D. Rorabaugh. A bound on a convexity measure for point sets. Preprint, `arXiv:1409.4344`, 2014.

[RR94] R. Ronfard and J. Rossignac. Triangulating multiply-connected polygons: A simple yet efficient algorithm. *Comput. Graph. Forum*, 13:C281–292, 1994.

[Sei91] R. Seidel. A simple and fast incremental randomized algorithm for computing trapezoidal decompositions and for triangulating polygons. *Comput. Geom.*, 1:51–64, 1991.

[She92] T.C. Shermer. A linear algorithm for bisecting a polygon. *Inform. Process. Lett.*, 41:135–140, 1992.

[SSW13] M. Sharir, A. Sheffer, and E. Welzl. Counting plane graphs: perfect matchings, spanning cycles, and Kasteleyn's technique. *J. Combin. Theory Ser. A*, 120:777–794.

[ST94] M. Sharir and S. Toledo. External polygon containment problems. *Comput. Geom.*, 4:99–118, 1994.

[Toe11] O. Toeplitz. Über einige Aufgaben der Analysis situs. *Ver. Schweiz. Nat. Gesel. Solothurn*, 4:197, 1911.

[TTW11] C.D. Tóth, G.T. Toussaint, and A. Winslow. Open guard edges and edge guards in simple polygons. In *Proc. XIV Spanish Meeting Comput. Geom.*, vol. 7579 of *LNCS*, pages 54–64, Springer, Berlin, 2011.

Preliminary version (July 25, 2017). To appear in the Handbook of Discrete and Computational Geometry,
J.E. Goodman, J. O'Rourke, and C. D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.