Approximate Unions of Lines and Minkowski Sums

Marc van Kreveld

A. Frank van der Stappen

institute of information and computing sciences, utrecht university

technical report UU-CS-2004-061

www.cs.uu.nl

Approximate Unions of Lines and Minkowski Sums

Marc van Kreveld
marc@cs.uu.nl

A. Frank van der Stappen
frankst@cs.uu.nl

Institute of Information and Computing Sciences,
Utrecht University, The Netherlands

Abstract

We study the complexity of and algorithms to construct approximations of the union of lines and of the Minkowski sum of two simple polygons. We also study *thick* unions of lines and Minkowski sums, which are inflated with a small disc. Let $b = D/\varepsilon$ be the ratio of the diameter of the region of interest and the maximum distance (or error) of the approximation. An approximate union of lines or Minkowski sum has complexity $\Theta(b^2)$ in the worst case. The thick union of n lines has complexity $\Omega(n + b^2)$ and $O(n + b\sqrt{bn})$, and thick Minkowski sums have complexity $\Omega(n^2 + b^2)$ and $O((n + b)n\sqrt{b \log n} + n^2 \log n)$ in the worst case. We present algorithms that run in $O(n + n^{2/3+\delta} b^{4/3})$ and $O(\min(bn,\, n^{4/3+\delta} b^{2/3}))$ time (any $\delta > 0$) for approximate and thick arrangements. For approximate Minkowski sums, the running time is $O(\min(b^2 n,\, n^2 + b^2 + (nb)^{4/3+\delta}))$; thick Minkowski sums take $O(n^{8/3+\delta} b^{2/3})$ time to compute.

1 Introduction

Minkowski sums are a basic concept in motion planning and therefore, they have been studied extensively in computational geometry. For two sets P and Q in the plane, it is defined as $P \oplus Q = \{p + q \,|\, p \in P,\, q \in Q\}$, where p and q are points whose coordinates are added, see Figure 1 for an example. Minkowski sums are used to define the free configuration space of a translating robot amidst obstacles [15, 22]. For essentially the same reason, Minkowski sums are important in packing problems, where given shapes have to be packed without overlap and without rotation inside a given region. This problem shows up in efficient use of fabric in cloth manufacturing [4, 13, 17]. Other applications are in cartography [24]. One of the basic questions that arises is: Given two simple polygons with n vertices in total, compute their Minkowski sum. All algorithms for this problem must take $\Omega(n^4)$ time, simply because the output can have as large complexity.

In practice, Minkowski sums of high complexity rarely show up, and it is interesting to discover under what conditions the maximum complexity of the Minkowski sum is considerably lower. Similarly, it is interesting to know when we can compute it more efficiently. Questions like these have led to research on *realistic input models* and the development of geometric algorithms for *fat objects*. One of the main results is that the union complexity of n triangles, all of whose angles are at least some given constant, is $O(n \log \log n)$ rather than $\Theta(n^2)$ [16, 20]. More recently, Erickson showed that two *local polygons* with n vertices have a Minkowski sum of complexity $O(n^3 \log n)$ [10]. A lot of research has been done on realistic input models, see [5, 6, 7, 8, 9, 18, 23, 25] for a sample.

This paper also deals with Minkowski sums and ways to overcome the $\Omega(n^4)$ lower bound for construction. We will not make assumptions on the input polygons, but instead study *approximate* Minkowski sums. We allow the Minkowski sum that we compute to be off by a distance ε, and assume that there is a region of interest with diameter D. We let $b = D/\varepsilon$ be the relative precision, and we prove complexity and running time bounds for approximate Minkowski sums. We assume

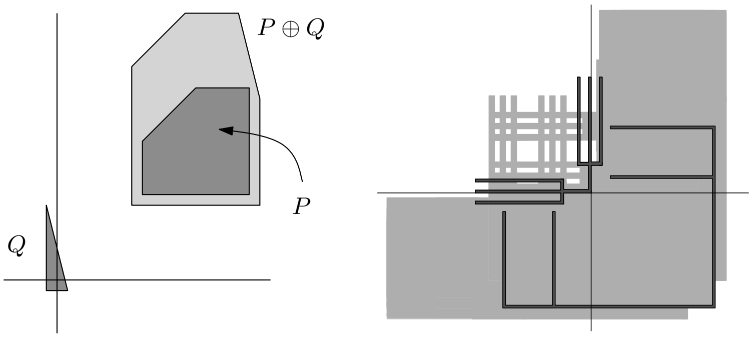

Figure 1: Left: Minkowski sum of P and Q. Right: The Minkowski sum of two polygons can have complexity $\Theta(n^4)$ in the worst case.

throughout the paper that $b = \Omega(1)$. If we do not make assumptions on both ε and D, it is impossible to show any bound better than $\Theta(n^4)$.

As a simpler first step we consider approximate unions of lines. We show that a union of n lines admits an approximation of complexity $O(\min(b^2, n^2))$, which is optimal in the worst case, and we can compute it in $O(n + n^{2/3+\delta} b^{4/3})$ time assuming $b = O(n)$, for any $\delta > 0$. A special type of approximate union is the union of thick lines, or strips. It is the maximal subset of the plane that is approximate. Being off by a distance at most ε implies that the strips have width 2ε. We show that n such strips in a region of diameter D have union complexity $\Omega(n + b^2)$ and $O(n + b\sqrt{bn})$ in the worst case, and we can compute it in $O(\min(bn,\, n^{4/3+\delta} b^{2/3}))$ time assuming $b = O(n)$.

Then we proceed with approximate and thick Minkowski sums. For thick Minkowski sums we prove complexity bounds of $\Omega(n^2 + b^2)$ and $O((n + b)n\sqrt{b \log n} + n^2 \log n)$ in the worst case. The approximation can be computed in $O(\min(nb^2,\, (nb)^{4/3+\delta}))$ time and the thick Minkowski sum in $O(n^{8/3+\delta} b^{2/3})$ time.

Interestingly, papers on fatness also imply complexity bounds on thick arrangements and Minkowski sums [8, 16, 20]. The fatness parameter is related to b and we would get a bound of $O(nb \log b)$ for the thick union of lines and $O(n^2 b^2 \log b)$ for the thick Minkowski sum [20]. The bounds presented in this paper are better for most values of b: For any $\gamma > 0$, we show better bounds if $b = \Omega(n^\gamma)$ and $b = O(n^{1-\gamma})$ (thick union of lines), or if $b = \Omega(n^\gamma)$ and $b = O(n^{2-\gamma})$ (thick Minkowski sums).

In the next section we give precise definitions of approximate and thick planar sets, and identify the interesting cases in terms of precision. Section 3 deals with approximate and thick union of lines, and Section 4 deals with approximate and thick Minkowski sums. Section 5 gives conclusions and open problems.

2 Approximate and thick sets

In this section we define approximate and thick sets in the plane.

Definition 1 *An ε-approximation of a set $C \subseteq \mathbf{R}^2$ is a set $A \subseteq \mathbf{R}^2$ such that for any point $p \in A$, there is a point $p' \in C$ at a distance at most ε, and for any point $p \in C$, there is a point $p' \in A$ at a distance at most ε.*

We will also study a particular ε-approximation. The *ε-thick set C* is the largest-area ε-approximation of C. Let Δ_ε be the disc of radius ε centered at the origin.

Definition 2 *The ε-thick set $C \subseteq \mathbf{R}^2$ is the Minkowski sum $C \oplus \Delta_\varepsilon$.*

We are interested in ε-approximations of reduced boundary complexity. A small absolute precision value ε, however, does not lead to a complexity reduction. Let $I \subseteq \mathbf{R}^2$ be a region of interest, and let D be its diameter. The interesting cases in terms of complexity reductions occur when the ratio of the diameter D of I and the precision value ε is bounded. The ratio of $b = D/\varepsilon$ can be seen as a measure for (the inverse of) the relative precision of the approximation.

3 Unions of lines

Let L be a set of n lines. We study the combinatorial complexity of approximate and thick unions of the lines in L. In addition we present algorithms for computing the approximate and thick unions. We assume for simplicity that the region of interest I is an axis-aligned square of diameter D.

3.1 Complexity

The complexity of an arrangement of n lines is $O(n^2)$ and this bound is tight. Approximations of lower complexity exist for the union of the n lines in L when the relative precision $b = o(n)$. We first prove a lower bound.

Lemma 1 *Any polygonal ε-approximation of the union of n lines has complexity $\Omega(b^2)$ in the worst case, assuming $b = O(n)$.*

Proof: Place a set L of $O(b)$ horizontal and vertical lines at distance 6ε to form a regular square grid in R, see Figure 2, and consider any approximate union U. All grid cells contain a square of size 4ε by 4ε (shown in grey in Figure 2, left) that cannot have any point in U. Besides these empty squares there are also squares of size 2ε by 2ε—both horizontally and vertically bisected by lines from L—and rectangles of size 2ε by 4ε—either horizontally a vertically bisected by a line from L—that must contain some point in U. Consider the set S of horizontal and vertical segments of length 2ε that bound the squares of size 2ε by 2ε, and move them $\varepsilon/2$ away from the intersections of L (segments of S are dashed in Figure 2, right). As a result, the rectangles between two empty (grey) squares now get size 2ε by 3ε. Each such rectangle must contain some point (not necessarily a vertex) of U. We claim that every group of two by two grid cells must contain a vertex of U. Let C, C', and C'' be the empty (grey) squares in the upper left, upper right, and bottom left grid cells of such a group. Let R_w and R_n be the 2ε-by-3ε rectangles below and to the right of C respectively. Let R_e and R_s be the 2ε-by-3ε rectangles below C' and to the right of C'' respectively. Finally, we let M be the twelve-gon between R_w, R_n, R_e, and R_s. If R_w or R_n contains some vertex of U we are done. Otherwise, these rectangles must be crossed

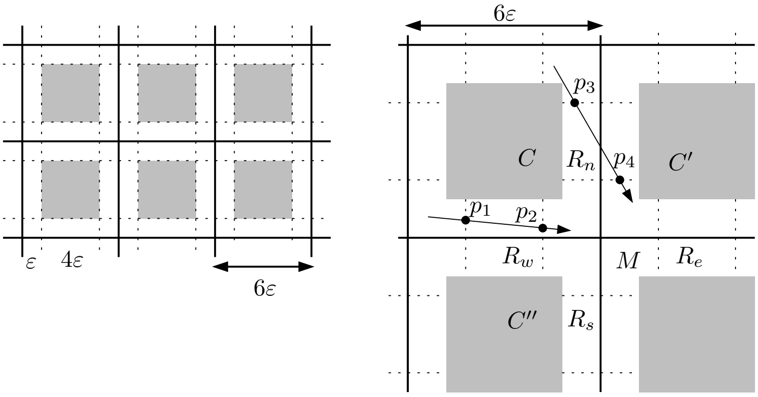

Figure 2: $\Omega(b^2)$ lower bound construction.

completely by edges of U. Consider the edges of U closest to C; they intersect the segments of S in points p_1, p_2, p_3, and p_4 as shown in the figure. Now either the edge of U through p_1 and p_2 has its right endpoint in $M \cup R_e \cup R_s$, or the edge of U through p_3 and p_4 has its lower endpoint in $M \cup R_e \cup R_s$. In summary, the symmetric *plus*-shaped region $M \cup R_e \cup R_s \cup R_w \cup R_n$ must contain a vertex. Since there are $\Omega(b^2)$ groups of two by two grid cells, any union U must have $\Omega(b^2)$ vertices. □

Lemma 2 *Any union of n lines has an ε-approximation of size* $O(\min(b^2, n^2))$.

Proof: As the set L of lines is its own ε-appoximation, the $O(n^2)$ bound is trivial. Now consider the b^2 points on a regular $\varepsilon \times \varepsilon$ grid inside I. Any point closer than ε from some point on a line in L is chosen to be in a set A. It is clear that A is an ε-approximation and of complexity $O(b^2)$. □

The ε-thick union of lines equals the union of ε-thick lines. Let $L_\varepsilon = \{l \oplus \Delta_\varepsilon \,|\, l \in L\}$. All elements of L_ε are strips of width 2ε.

Lemma 3 *The complexity of the ε-thick union of n lines can be $\Omega(n + b^2)$.*

Proof: Choose $b/3$ equally-spaced horizontal strips (e.g. at distance ε apart) that all intersect I. Choose $b/3$ vertical strips in the same way. The complexity of the union of these strips is $\Omega(b^2)$. We let $\Omega(n)$ strips with slightly different positive slopes intersect the uncovered lower right corner of I, such that each of the $\Omega(n)$ pairs of strips with successive slopes adds a vertex on an arbitrarily small circular arc centered at the corner. □

A hole in the union of strips from L_ε originates from a hole with area $\Omega(\varepsilon^2)$ in the union of the lines from L. As a result, the number of holes in the union of strips in I is $O(b^2)$. This bound plays a role in the proof of the following result.

Lemma 4 *The complexity of the ε-thick union of n lines is $O(n + b\sqrt{bn})$.*

Proof: A vertex v of the union is the intersection of a line β_i bounding a strip $s_i \in L_\varepsilon$ and a line β_j bounding another strip $s_j \in L_\varepsilon$. It is an endpoint of the equally-long segments $\beta_i \cap s_j$ and $\beta_j \cap s_i$, whose interiors are in the interior of the union. We separate the vertices v inside I for which $\beta_i \cap s_j$ intersects the boundary ∂I of I from those for which this segment lies entirely inside I.

If $\beta_i \cap s_j$ intersects ∂I, we charge the vertex v to the intersection $\beta_i \cap s_j \cap \partial I = \beta_i \cap \partial I$. The intersection will be charged only once, giving $O(n)$ vertices of this type in total.

We next handle the union vertices v in I for which $\beta_i \cap s_j$ lies entirely inside I. Each union vertex v belongs to a convex hole. We define the turning angle α_v to be $\pi - \phi$, where ϕ $(0 < \phi < \pi)$ is the angle between the union edges incident to v. Let $\gamma = \sqrt{b/n}$. We bound the number of vertices v with $\alpha_v > \gamma$ differently from those with $\alpha_v \leq \gamma$.

Observe that the sum of the turning angles of the vertices of a single hole equals 2π. Since the number of holes in I is $O(b^2)$, the sum of the turning angles of all union vertices in I is $O(b^2)$ as well. The number of union vertices v with $\alpha_v > \gamma$ inside I is therefore $O(b^2/\gamma) = O(b\sqrt{bn})$.

Observe that the length of the part of each bounding line of a strip inside I is at most D. Since the number of strips intersecting I is $O(n)$, the total length of all bounding lines in I is $O(Dn)$. The intersection of the strips s_i and s_j causes the interior of the segment $\beta_i \cap s_j$ adjacent to v to be in the interior of the union. For a vertex v with $\alpha_v \leq \gamma$ this means that a stretch of length $|\beta_i \cap s_j| = 2\varepsilon / \sin \alpha_v \geq 2\varepsilon/\alpha_v \geq 2\varepsilon/\gamma$ on β_i adjacent to v cannot contain another union vertex. The number of union vertices v with $\alpha_v > \gamma$ inside I is therefore $O(nD/(\varepsilon/\gamma)) = O(b\sqrt{bn})$. □

Remark. Another way of bounding the complexity of the thick union is by many-faces bounds in arrangements of lines [21, 12]. We are interested in $O(b^2)$ faces (the number of holes) in an arrangement of $O(n)$ lines, which yields a bound of $O(n + n^{2/3}b^{4/3})$. This bound is slightly worse than our bound.

3.2 Algorithms

As we observed, an approximate union of lines need only be a set of $O(b^2)$ points (if $b = O(n)$) that is a subset of a regular square grid with spacing ε inside I. For every point in this grid we need to find out whether it lies inside at least one of the n strips with width 2ε, centered at the input lines. Every grid point inside some strip is guaranteed to be within ε of a line, and every point on an input line is guaranteed to be within $\varepsilon/\sqrt{2}$ from a covered grid point. We can get the covered grid points in several ways. The easiest is to iterate over the strips, and for each strip find the $O(b)$ points inside. This takes $O(nb)$ time in total. In case we want to find a superset of the union of lines which is also an approximate union, we can take groups of four grid points covered by the same strip and put the whole grid square in the approximate union. For groups of three we take the triangle that is half a grid square.

A more efficient algorithm for larger values of b is obtained by building a data structure on the strips, such that for a query point, we find out efficiently whether it lies inside some strip. Using partition trees, we can make an $O(m)$ size data structure in $O(m^{1+\delta})$ time such that queries can be answered in $O(n^{1+\delta}/\sqrt{m})$ time, for any $n \leq m \leq n^2$ [1]. We will query with the $O(b^2)$ grid points, so we choose m to balance the preprocessing time and the total query time. The resulting algorithm takes $O(n + n^{2/3+\delta}b^{4/3})$ time.

Theorem 1 *An ε-approximate union of n lines inside a square of diameter D has complexity $O(b^2)$ and can be computed in $O(nb)$ and in $O(n + n^{2/3+\delta}b^{4/3})$ time, for any $\delta > 0$, and where $b = D/\varepsilon = O(n)$.*

The union of strips is one of the problems known to be 3SUM-hard [11], implying that a subquadratic time algorithm for the construction is not likely to exist. But since we have ε-thick strips and are interested in a region of diameter D only, we can achieve subquadratic construction time.

To compute the union of n ε-thick strips we will make use of multi-level partition trees and ray shooting [1]. All edges that appear in the thick union are segments on the set L of $2n$ lines bounding the strips; we call them *exposed edges*. We will shoot along the lines bounding the strips to find these exposed edges. We build two data structures: one for querying when we are outside the union of the strips (to find the end of an exposed edge), and one for querying when we are inside the union of the strips (to find the start of an exposed edge). The former structure is easy. We simply preprocess all lines L bounding strips into a ray shooting structure.

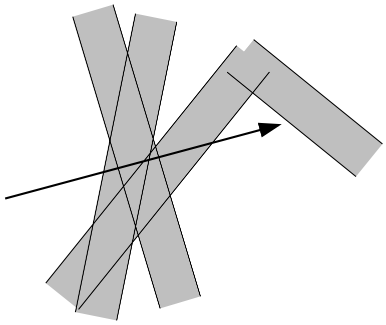

Figure 3: Ray shooting to find exposed edges.

It is more difficult to find the start of an exposed edge. When the starting point of the query ray is inside the union of the strips, we do not want to find the next intersection with some strip boundary, because it may be covered by a third strip and hence, not be part of the output. The region I may contain $\Theta(n^2)$ intersection points of bounding lines in total. Instead of normal ray shooting, we build a partition tree T of which the main tree allows us to select all strips that contain the query point q (start of the ray) represented by a small number of canonical nodes. For every node in the main tree we build an associated structure for ray shooting in the represented strips, *but from infinity and in the opposite direction.* This way we find the point q' where the ray will exit all strips in which the query point q lies. Either q' is a vertex of the union of strips and we start to find an exposed edge, or q' lies inside one or more strips, which necessarily are different from the ones that contained q before. We use one more query in T to distinguish between these cases. If q' lies inside, we continue to find the last exit point of the strips that contain q', like we did before for q. It is clear that this approach will (eventually) find the next starting point of an exposed edge along the line we are processing.

Lemma 5 *Let q be any point inside the union of strips, and assume we find a point q' due to the query in T. If the point q' lies in the union of strips, then the next point q'' we find after querying in T has distance $\geq \varepsilon$ from q.*

Proof: Suppose we query with q and find that q' is in the union. Then q' must lie in some strip s_i in which q does not lie. Suppose we query with q' and find a point q''. Point q'' must lie outside s_i, so $\overline{qq''}$ cuts through S completely. This proves the lemma. □

The lemma shows that the number of ray shooting queries inside the union is at most $O(b)$ for any line, so $O(nb)$ for all lines of L together. We also do at most $O(n + b\sqrt{bn})$ queries outside the union due to the bound on the number of exposed edges. Since $b = O(n)$, we do $O(nb)$ ray shootings. For both structures we have $O(m^{1+\delta})$ preprocessing time and $O(n^{1+\delta}/\sqrt{m})$ query time by the standard theory of partition trees [1]. We balance the preprocessing and the total query time and obtain an $O(n^{4/3+\delta}b^{2/3})$ time bound to determine all exposed edges. It is easy to assemble these into the union of strips without affecting the time bound asymptotically.

Theorem 2 *The union of a set of n ε-thick strips inside a square of diameter D has complexity $O(n+b\sqrt{bn})$ can be constructed in $O(n^{4/3+\delta}b^{2/3})$ time, for any $\delta > 0$, and where $b = D/\varepsilon = O(n)$.*

Remark. An ε-thick strip inside a square of side D can be seen as the union of two b-fat wedges, in the definition of [8], or two b-fat triangles in the definition of [16]. The former paper implies an $O(nb^2 \log b)$ complexity bound and an $O(nb^2 \log b \log(bn))$ time algorithm, both considerably worse than our bounds unless b is very small with respect to n. Similarly, if a strip is treated as the union of $O(b)$ squares, then we can use Efrat's result [6] to bound the union of $O(bn)$ fat squares by $O(\lambda_{10}(bn))$.

4 Minkowski sums

Let P and Q be polygons with n vertices. We study the combinatorial complexity of approximate and thick Minkowski sums $P \oplus Q$. In addition we present algorithms for computing approximate and thick Minkowski sums. We assume that both polygons have diameter at most D, so that the Minkowski sum $P \oplus Q$ has diameter $O(D)$.

4.1 Complexity

The complexity of the Minkowski sum of two polygons with n vertices is $O(n^4)$ and this bound is tight in the worst case. The following lemma says that approximations of lower complexity exist when the relative precision $b = o(n^2)$. We omit the proof as it is similar to that of Lemma 2. Tightness of the bound follows if we use polygons whose Minkowski sum generates a grid similar to the one in Figure 2. Figure 1 shows that such a grid can easily be made.

Lemma 6 *Any Minkowski sum of two polygons with n vertices admits an ε-approximation of complexity $O(\min(b^2, n^4))$, and this bound is tight in the worst case.*

We turn our attention to the thick Minkowski sum.

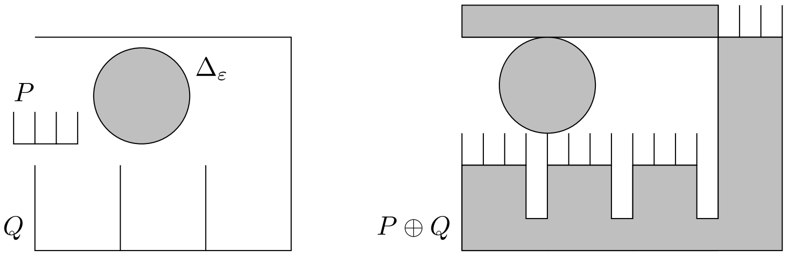

Figure 4: The Minkowski sum of P, Q and an ε-disc Δ_ε can have $\Omega(n^2)$ holes.

Lemma 7 *The complexity of the ε-thick Minkowksi sum of two polygons with n vertices can be $\Omega(n^2 + b^2)$.*

Proof: The bound of $\Omega(b^2)$ is trivial. Figure 4 shows that $P \oplus Q \oplus \Delta_\varepsilon$ can have $\Omega(n^2)$ holes. The sum of the lengths of the $\Theta(n)$ spikes of P and of Q is such that the distance from the $\Theta(n^2)$ resulting spikes of $P \oplus Q$ to the horizontal top bar of $P \oplus Q$ is 2ε. The Minkowski sum of $P \oplus Q$ with Δ_ε will have one tiny hole between each of the $\Theta(n^2)$ pairs of supporting lines of consecutive spikes of $P \oplus Q$, also bounded by the horizontal edge at the top of $P \oplus Q$. □

We deduce an upper bound on the complexity of the thick Minkowski sum by considering the complexity of the union boundary of all Minkowski sums of an edge of P, a vertex of Q, and Δ_ε, and all Minkowski sums of an edge of Q, a vertex of P, and Δ_ε. The boundary of the union of these racetracks (which are in fact thick translated edges of P or Q) is a superset of the boundary of $P \oplus Q \oplus \Delta_\varepsilon$.

Let C be the set of $O(n^2)$ racetracks. A racetrack is *short* if its diameter is at most 4ε and *long* otherwise. Let C_s and C_l be the subsets of short and long racetracks in C respectively. We regard each racetrack as the union of two discs of diameter 2ε and a rectangle of width 2ε. Let S be the set of all discs resulting from racetracks in C. Let R_l be the set of rectangles resulting from the racetracks in C_l. For each racetrack we also use a hexagonal inner approximation of it by replacing both discs by enclosed squares with side lengths $\varepsilon\sqrt{2}$. The hexagonal racetrack is mirror-symmetric in two lines, and has four sides of length $\varepsilon\sqrt{2}$. Observe that each hexagonal racetrack is the Minkowski sum of a translated copy e of an edge of P or Q with a square with side length $\varepsilon\sqrt{2}$ whose diagonal is orthogonal to e. Let H_s be the set of hexagonal racetracks resulting from the racetracks in C_s.

It is well known that the union complexity of a collection of unit discs is linear in the number of discs [21]. The complexity of $\bigcup_{s \in S} s$, and hence the number of vertices resulting from pairwise intersections of circular arcs in C, is therefore $O(n^2)$. For succinctness of notation, we let $\bigcup X = \bigcup_{x \in X} x$ and $\bigcup X, Y = (\bigcup_{x \in X} x) \cup (\bigcup_{y \in Y} y)$. We derive a bound on the number of remaining vertices of the union $\bigcup C = \bigcup C_s, C_l$ of racetracks in three steps. We will bound

- the number of vertices of $\bigcup C$ resulting from pairwise intersections of straight segments by considering the number of vertices in $\bigcup R_l, H_s$,
- the number of vertices of $\bigcup C$ resulting from intersections of a circular arc and a straight segment by separately considering the number of vertices in $\bigcup R_l, S$ and the number of vertices in $\bigcup C_s, S$.

We do not use short rectangles in the proofs. They have two sides of length 2ε, but the other sides can be arbitrarily short. We concentrate on bounding the complexity of $\bigcup R_l, H_s$ first.

Lemma 8 *The number of holes in the union $\bigcup R_l, H_s$ is $O(n^2 + b^2)$.*

Proof: Consider the hexagons and rectangles, and shrink them to dumbbells of one of two types, see Figure 5. Shrinking can decrease the number of holes, and only when two holes (faces) merge

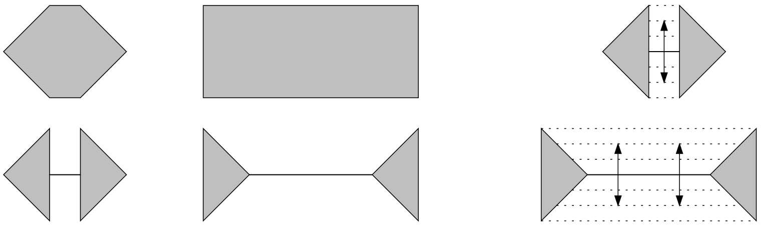

Figure 5: A dumbbell for a hexagon and for a rectangle.

into one. When this happens, an original vertex of a hexagon or rectangle appears in the union of the shrunk shapes, and this can happen only once for each vertex (since the vertices themselves do not move). Therefore, the reverse process of growing dumbbells to rectangles and hexagons can create at most $O(n^2)$ new holes. (This argument was used in [16] to transform triangles into canonical form.)

It remains to analyse how many holes the $O(n^2)$ dumbbells make, and how many can survive the growing process. First, consider the set T of $O(n^2)$ equi-angular right-angled triangles of the same size at the ends of the dumbbells. The union complexity and number of holes of these triangles is $O(n^2)$ by Matoušek et al. [16]. Let E be the set of $O(n^2)$ middle edges of the dumbbells, they can be the cause of many more holes. Take any hole of $\bigcup T, E$. There are $O(n^2)$ holes that contain a vertex of $\bigcup T$. All other holes are convex and must be bounded by at least two edges from E. If a hole is bounded by edges of E only, then it will close if it has area less than ε^2, see Figure 6. We extend this argument to account for the edges of $\bigcup T$, which do not grow.

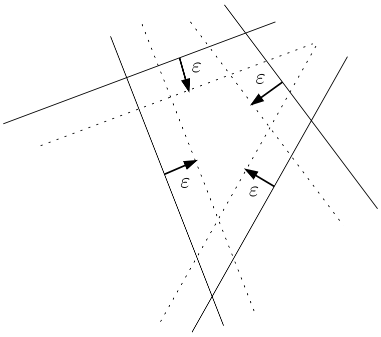

Figure 6: A hole with edges of $\bigcup E$.

Consider $\bigcup E$. It has $O(b^2)$ convex holes that can survive the growing by the arguments already given. Now add the edges of $\bigcup T$. There are only $O(n^2)$ holes that contain a vertex of $\bigcup T$. We claim that all other edges of $\bigcup T$ will be covered completely after growing the edges of E, and hence do not influence the number of holes. Indeed, any edge of $\bigcup T$ has length at most $\varepsilon\sqrt{2}$, so in any hole where it does not end, it intersects two edges of E. These edges will grow and cover everything within distance $\varepsilon/\sqrt{2}$ that is not yet covered by $\bigcup T$, which can easily be seen from the dumbbell shapes. □

Note that the proof depends on the fact that the objects are polygonal, and not curved like the racetracks themselves. We do not have a similar bound on the number of holes of $\bigcup C$. The bound on the number of holes of $\bigcup R_l, H_s$ plays a role in the following result.

Lemma 9 *The complexity of the union $\bigcup R_l, H_s$ is $O((n+b)n\sqrt{b\log n} + n^2\log n)$.*

Proof: Since the number of holes of the union is $O(n^2 + b^2)$ and the number of convex vertices of the union is $O(n^2)$, the total turning angle of all intersection points of a rectangle or hexagon and another rectangle or hexagon that appear on the union is also $O(n^2 + b^2)$. We bound the intersections with angle $> \gamma$ separately from the ones with angle $\leq \gamma$, like in Lemma 4. If $b = O(n)$, we let $\gamma = 1/\sqrt{b \log n}$, and we have $O(n^2\sqrt{b \log n})$ vertices with angle $> \gamma$. If $b = \Omega(n)$, we let $\gamma = \sqrt{b/\log n}/n$ and we have $O(nb\sqrt{b \log n})$ vertices with angle $> \gamma$.

It remains to handle the intersection points where a rectangle or hexagon intersects another rectangle or hexagon with a turning angle $\leq \gamma$. We use a combination of the ideas from Lemma 4, a divide-and-conquer scheme, and charging to the endpoints of the rectangles.

Consider the vertices of the $O(n^2)$ rectangles of R_l and hexagons of H_s. Draw a vertical line through the leftmost and the rightmost vertex to start the divide-and-conquer. We repeatedly choose a vertical line through the vertex with median x-coordinate in the vertical strip, until no vertex is inside any strip. For any vertical strip V in the process, we call a rectangle or hexagon *short* if it has at least one vertex inside V. We call a rectangle or hexagon *long* if none of its vertices lie in the strip, but it was short at the (parent) strip from which V was created (compare this with segment trees, in particular hereditary segment trees [3]). Every rectangle and every hexagon is short at $O(\log n)$ strips and long at $O(\log n)$ strips as well.

Every intersection of a rectangle or hexagon with another rectangle or hexagon appears as a long-long intersection or as a long-short intersection in a strip. Any short-short intersection will be counted later in the recursive process, closer to the leaves, where it will be a long-long or long-short intersection [3].

For any strip V we analyse how many union vertices it contains. The long-long intersections are analyzed as in Lemma 4: Consider a side s of a long rectangle or hexagon o and its union vertices where the turning angle is $\leq \gamma$ (the angle of the complement of the union is $\geq \pi - \gamma$). First consider the union vertices on s with other long rectangles and hexagons. Then either side s is covered over a stretch $\varepsilon\sqrt{2}/\gamma$, or the covered part of s is incident to one of the vertical lines bounding V. In the latter case we charge the intersection to o, which will be charged at most 4 times in V this way. In the former case we are in the same situation as Lemma 4. Second, consider the union vertices on a side s of a short rectangle or hexagon o_s with long rectangles and hexagons. Let o_l be a long rectangle or hexagon that intersects s with an angle $\leq \gamma$ and the intersection is a union vertex v. If s intersects two parallel sides of o_l in V, then the stretch of s inside o_l is at least $\varepsilon\sqrt{2}/\gamma$. If s intersects only one side of o_s, then the union vertex v is the first one along s when seen from one of the vertical strip boundaries, or from an endpoint of s that lies inside o_l. In both cases we charge it to s. Any side of a rectangle or hexagon is charged $O(1)$ times in a strip, plus at most $D/(\varepsilon/\gamma) = b\gamma$ times due to length of covering. Summing over all sides and all strips yields the stated bounds. □

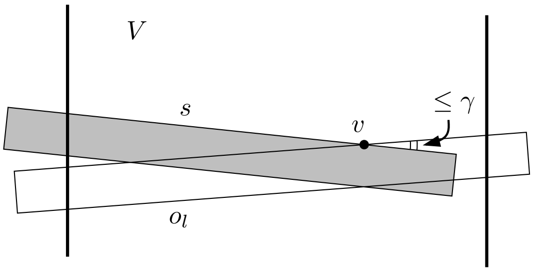

Figure 7: A side s of a short rectangle and its intersection with a long rectangle in strip V.

The following interesting theorem combines the union complexity of discs and rectangles. It is a variation on a result by Pach and Sharir [19]. We emphasize that a unit-width rectangle is a rectangle whose shortest side has unit length. The theorem provides the key to our bound for the complexity of $\bigcup R_l, S$.

Theorem 3 *Given a set R of m unit-width rectangles with union complexity $O(f(m))$ and a set S of n unit-diameter discs. Then the complexity of union of the rectangles and discs, $\bigcup R, S$, is $O(f(m)+n)$.*

Proof: We call any intersection point between discs and/or rectangles *exposed* if it appears as a vertex in $\bigcup R, S$, and otherwise it is *covered*.

We will charge the exposed intersection of any disc and rectangle to: (i) a vertex of a rectangle; (ii) an intersection point of two rectangles that is exposed in $\bigcup R$; (iii) a disc; (iv) an intersection point of two discs that is exposed in $\bigcup S$. Let d be any disc whose exposed intersection points we wish to charge. Consider an exposed arc a of d ending at exposed vertices u, v. If u or v is the intersection of d with some other disc, then we charge both u and v to the exposed disc-disc intersection. So assume from now on that u is an intersection point with a rectangle r_i and v is an intersection point with a rectangle r_j, see Figure 8.

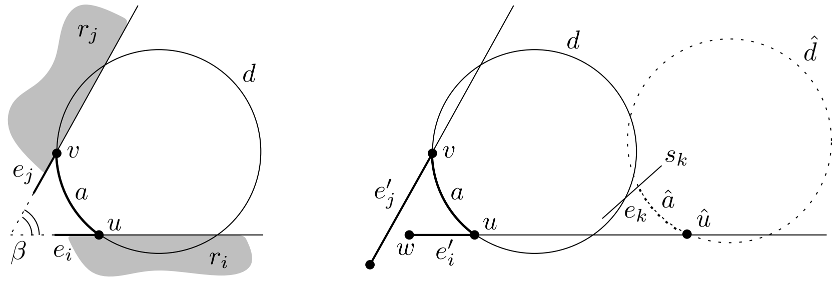

Figure 8: Illustration of the proof.

If r_i (or r_j) covers an arc of d of length at least $\pi/6$ then we charge the exposed arc a to d itself. Disc d can be charged at most $O(1)$ times in this way. Similarly, if a itself has length at least $\pi/6$, then we charge a to d itself. Note that this captures the case where $r_i = r_j$ too, due to the unit-width and unit-diameter conditions. Also observe that if a unit-diameter disc d intersects a unit-width rectangle r in more than two points, then r must cover some arc of length greater than $\pi/6$ of d. As a result, we can now assume that rectangles r_i and r_j cover only one arc of d.

The sides of r_i and r_j that contain u and v are denoted s_i and s_j, respectively, and the exposed edges of $\bigcup R, S$ on s_i and s_j ending at u and v are denoted e_i and e_j. If an endpoint of s_i or s_j lies inside d, then we charge a to r_i or r_j. Vertex u or v is then the first exposed vertex along s_i or s_j, counting from the endpoint inside d.

If e_i and e_j meet, or we extend them such that they meet, then they make an angle of $\beta \geq \pi/3$ due to the maximum length of $\pi/6$ for a and the adjacent covered arcs of d, and the other conditions (unless we can already charge a to something), see Figure 8. Consider $\bigcup R, \{d\}$, and let e'_i and e'_j be edges of this union that contain or are equal to e_i and e_j, respectively. Due to symmetry we can assume without loss of generality that $|e'_i| \leq |e'_j|$. Then $|e'_i| \leq 1/2$ due to the angle β and the unit-size objects. Let w be the other exposed vertex of e'_i, that is, $\overline{wu}$ is an exposed edge of $\bigcup R, \{d\}$. Vertex w is also exposed in $\bigcup R$. Then we charge arc a to w. There cannot be another arc like a that must be charged in the same way to w. To see this, assume that another disc $\hat{d}$ must charge an arc $\hat{a}$ to w for the same situation. Then $\hat{u} = s_i \cap \hat{d}$ (the intersection point closer to w) must have distance at most $1/2$ from w, so d and $\hat{d}$ must intersect. Referring to Figure 8 (right), the other rectangle r_k must have its side s_k below the lower intersection point of d and $\hat{d}$ to allow that $\hat{a}$ has as an endpoint an exposed vertex on s_k, resulting in an edge e_k of $\bigcup R, S$. But due to the angle $\hat{\beta} \geq \pi/3$ that e_i and e_k make after extension, e'_k must be shorter than $\hat{e'}_i$ in $\bigcup R, \{\hat{d}\}$ so $\hat{a}$ cannot charge w in this way. This completes the charging scheme and the proof. □

Figure 9 shows a simple example that illustrates that the theorem is not true if discs can be arbitrarily large, or rectangles arbitrarily short. In both cases an $\Theta(nm)$ complexity construction can be made, while the union of the rectangles is only $O(m)$.

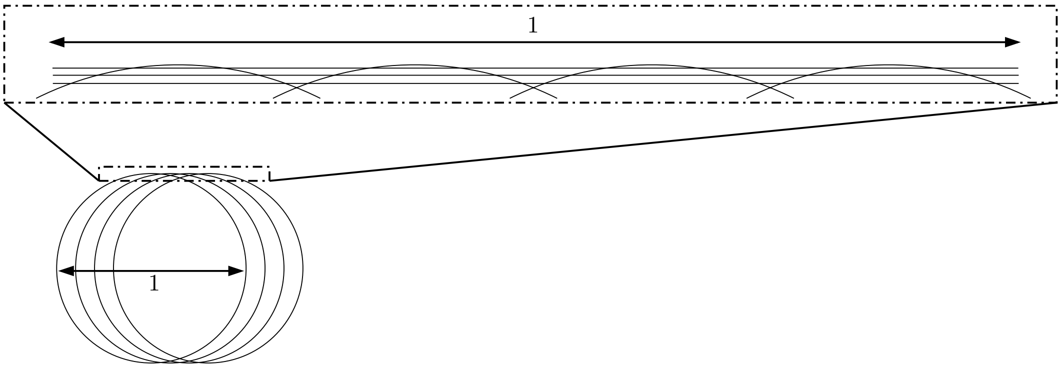

Figure 9: The points at the top represent very short vertical line segments. It illustrates that for the theorem to hold, rectangles should not be arbitrarily short.

Although Lemma 9 applies to the complexity of $\bigcup R_l, H_s$ it also implies a similar bound on the complexity of $\bigcup R_l$ alone. We now apply the result of Lemma 9 in Theorem 3 and obtain a complexity bound on the union of the rectangles in R_l and the discs in S, which leads to the following result.

Corollary 1 *The complexity of the union $\bigcup R_l, S$ is $O((n+b)n\sqrt{b \log n} + n^2 \log n)$.*

It remains to bound the complexity of $\bigcup C_s, S$. The shapes in $C_s \cup S$ are all fat and have comparable sizes. Let τ be an equilateral triangle whose sides have length $\varepsilon\sqrt{3}$. For each shape $f \in C_s \cup S$ and each point $p \in \partial f$, we can place τ fully inside f while one of its corners coincides with p. Moreover, we observe that the diameter of any $f \in C_s \cup S$ is between 2ε and 4ε. Efrat [6] states that in this case the union complexity is $O(\lambda_{s+2}(|C_s \cup S|)$, where s is the maximum number of boundary intersections of any two shapes from $C_s \cup S$.

Lemma 10 *The complexity of the unions $\bigcup C_s, S$ is $O(\lambda_6(n^2))$.*

The bounds on the complexities of the unions $\bigcup R_l, H_s$, $\bigcup R_l, S$, and $\bigcup C_s, S$ imply the following bound for the ε-thick Minkowski sum.

Lemma 11 *The complexity of the ε-thick Minkowski sum of two polygons with n vertices is $O((n+b)n\sqrt{b \log n} + n^2 \log n)$.*

4.2 Algorithms

The approximate Minkowski sum of two simple polygons can be computed in the same way as for approximate union of lines. We determine for each of the $O(b^2)$ points of an appropriately defined grid whether it is inside or outside the Minkowski sum. There are several ways to do this.

If b is small, we can test each grid point without preprocessing in $O(n)$ time. The test whether a given point—which we can assume to be the origin after a translation—lies in the Minkowski sum of two polygons P and Q reduces to the intersection test for two simple polygons P and $-Q$ (which is Q mirrored in the origin). This on its turn can be solved by testing if a (combined) polygon has self-intersections: compute a line segment s that connects the boundaries of P and $-Q$ without intersecting P and $-Q$ elsewhere, and convert s into a narrow passage to make a polygon that is a combination $\Gamma_s(P,Q)$ of P and $-Q$. The test whether polygon $\Gamma_s(P,Q)$ is simple determines whether P and Q intersect, and therefore whether the origin lies in the Minkowski sum of P and Q. The simplicity test takes linear time as a by-product of Chazelle's linear time triangulation algorithm of a simple polygon [2]. Concluding, we can compute an approximate Minkowski sum of two simple polygons with n vertices in total in $O(nb^2)$ time.

For larger values of b it becomes more efficient to see the Minkowski sum of P and Q as the union of $O(n^2)$ triangles. We can store them in a partition tree such that point containment queries can be answered efficiently [1]. Using the preprocessing versus total query time trade-off for range searching we obtain an $O(n^2 + b^2 + (nb)^{4/3+\delta})$ time algorithm for any constant $\delta > 0$. In the same way as for the approximate union of lines we can make the approximate Minkowski sum a connected subset of the plane.

Theorem 4 *An ε-approximate Minkowski sum of two simple polygons with n vertices in total, inside a square of diameter D has complexity $O(b^2)$ and can be computed in $O(nb^2)$ and in $O(n^2 + b^2 + (nb)^{4/3+\delta})$ time, for any $\delta > 0$, and where $b = D/\varepsilon = O(n^2)$.*

To compute the thick Minkowski sum of two simple polygons P and Q we use the ray shooting approach of Section 3. However, we cannot do exactly the same, because a ray that enters the union of rectangles and discs is not guaranteed to stay inside the union over a length at least ε. Furthermore, we have to deal with the circular parts of the thick Minkowski sum.

For convenience we will compute the Minkowski sum of every vertex of P, edge of Q, and Δ_ε, and every vertex of Q, edge of P, and Δ_ε. The union of these $O(n^2)$ racetracks is a subset of the thick Minkowski sum. It has the same asymptotic complexity, and from it the true thick Minkowski sum can be constructed easily by filling holes and removing boundary edges of these holes.

Let C again be the set of $O(n^2)$ racetracks. We consider each racetrack to be the union of two discs and a rectangle as before, giving a set S of $O(n^2)$ discs and a set R of $O(n^2)$ rectangles. The union of the discs has complexity $O(n^2)$ and can be constructed in $O(n^2 \log^2 n)$ time by the algorithm of Kedem et al. [14]. The union of the rectangles has complexity $O((n+b)n\sqrt{b \log n} + n^2 \log n)$, and we show next how to compute it. Recall that every rectangle has width 2ε, and the length can vary from nearly 0 to linear in D. Let R_s be the subset of (short) rectangles of length at most 4ε and let R_l be the rectangles of length $> 4\varepsilon$. For each rectangle in R_l, we cut off two squares of size 2ε by 2ε, one from either end. Let R'_l be the set of shortened rectangles from R_l, and let R'_s be the union of R_s and the $2|R_l|$ squares that were cut off.

We preprocess R'_s as follows. Compute the union of the rectangles in R'_s (through their hexagonal racetrack) explicitly with the algorithm of Kedem et al. in $O(\lambda_{14}(n^2)\log^2 n)$ time. It has complexity $O(\lambda_{14}(n^2))$ by Efrat's result [6]. (Two hexagons can intersect 12 times). Preprocess the edges of the boundary of the union into a ray shooting structure. The start of the query ray can be inside or outside the union.

We preprocess R'_l as in Section 3: the main tree is a partition tree in which all rectangles that contain a query point (start of the query ray) can be selected in a small number of canonical subsets, represented by nodes of the tree. For every node of the tree we store an associated structure for ray shooting 'from the other side', so we can find the last exit of the rectangles of R'_l.

A query is always performed in both structures. There are four cases for the start q of the query ray:

1. If $q \notin \bigcup R'_s$ and $q \notin \bigcup R'_l$, then the (double) ray shooting query will give the next vertex along the ray, which is in the union $\bigcup R$ as well.

2. If $q \in \bigcup R'_s$ and $q \notin \bigcup R'_l$, then either the query ray exits $\bigcup R'_s$ first, in which case we find a vertex of the union, or the query ray enters $\bigcup R'_l$ first. In the latter case, the next ray shooting query is done from the exit point of $\bigcup R'_s$.

3. If $q \notin \bigcup R'_s$ and $q \in \bigcup R'_l$, then either the query ray exits the subset of the rectangles of R'_l that contain it first, or the query ray enters $\bigcup R'_s$ first. In the latter case, the next ray shooting query can start at the exit of the subset from $\bigcup R'_l$. In the former case, we either exit $\bigcup R'_l$ and find a vertex of the union, or we stay inside $\bigcup R'_l$ and the next query will be of type 3 again.

4. If $q \in \bigcup R'_s$ and $q \in \bigcup R'_l$, then either the query ray exits $\bigcup R'_s$ last, or the query ray exits the subset of the rectangles of R'_l that contain it last. We will shoot from the furthest point hit, and the query can be of types 1, 2, 3, or 4.

We will perform the ray shooting queries along the sides of the rectangles in R, so we determine $\bigcup R$ this way. We can prove that ray shooting queries find a vertex of the union, or the starting point of a constant number of queries later, the query ray advances in a way similar to Lemma 5.

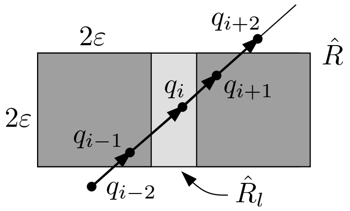

Figure 10: Advance of ray shooting when $q_i \in \bigcup R'_l$.

Lemma 12 *If five consecutive ray shooting queries stay fully inside $\bigcup R$, then the distance from the start of the first and end of the fifth ray is at least 2ε.*

Proof: Assume that five consecutive ray shooting queries $q_{i-2}, q_{i-1}, q_i, q_{i+1}, q_{i+2}$ (in both structures each) do not find any vertex of the union $\bigcup R$. Assume first that q_i is a starting point of a query ray that is inside some $\hat{R}_l \in R'_l$, where $\hat{R}_l$ was generated from a rectangle $\hat{R} \in R$. It is easy to see that the line supporting the query ray intersects $\hat{R}$ over a stretch of at least 2ε (Figure 10). It is also clear that q_{i+1}, the start of the next query ray, cannot be inside $\hat{R}_l$, and the same holds for q_{i-1}. The points q_{i-1} and q_{i+1} can be inside one of the end squares of $\hat{R}$, however, but q_{i-2} and q_{i+2} cannot be in the interior of $\hat{R}$. Hence, the advance from q_{i-2} to q_{i+2} achieved by four ray shootings in both structures is at least 2ε, or we have found a union vertex.

If q_i is not inside some $\hat{R}_l \in R'_l$ but is not part of the union, it lies inside some square or hexagon from R'_s. But since q_{i+1} is at the exit of $\bigcup R'_s$, it either is a union vertex or it lies inside some $\hat{R}_l \in R'_l$, and we can apply the same argument. □

We showed that the total number of queries done is bounded by $O(n^2 b)$ plus the union complexity of $\bigcup R$. Since we assume $b = O(n^2)$, we do $O(n^2 b + (n+b)n\sqrt{b \log n} + n^2 \log n)$ queries, and after balancing the preprocessing time and total query time we spend $O(n^{8/3+\delta} b^{2/3})$ time on ray shooting to compute $\bigcup R$.

After computing the union $\bigcup R$ of rectangles and the union $\bigcup S$ of discs we compute the union of unions with a simple overlay, for example by a plane sweep. Every new intersection point appears in the thick Minkowski sum, so the overlay step takes $O((n^2\sqrt{b}+nb\sqrt{b}+\lambda_{14}(n^2))\log^{2.5} n+n^2\log^3 n)$ time. This is always dominated by the total ray shooting time.

Theorem 5 *The ε-thick Minkowski sum of two simple polygons with n vertices in total inside a square of diameter D has complexity $O((n+b)n\sqrt{b \log n} + n^2 \log n)$ and can be constructed in $O(n^{8/3+\delta} b^{2/3})$ time, for any $\delta > 0$, and where $b = D/\varepsilon = O(n^2)$.*

5 Conclusions and open problems

We have studied the fundamental geometric concept of Minkowski sums in a context that bears resemblance to the study of realistic input models. We have presented new combinatorial and algorithmic results on approximate and thick unions of lines, and approximate and thick Minkowski sums. The results are related to fatness and realistic input models, but we take the fatness parameter into account explicitly. Although we studied unions of lines and Minkowski sums, our methods can be used for approximate and thick unions of triangles as well. For ε-approximate unions of n triangles in a region of interest with diameter D, the complexity is $\Theta(b^2)$ in the worst case, and we can compute it in $O(n + n^{2/3+\delta}b^{4/3})$ time, where $\delta > 0$ and $b = D/\varepsilon = O(n)$. For thick unions of triangles, we get an $\Omega(b^2 + n)$ and $O(n \log n + b\sqrt{bn \log n})$ complexity bound and an $O(n^{4/3+\delta}b^{2/3})$ time algorithm.

The main open problems are closing the gaps left in the complexity bounds and improving the running time of the algorithms. We believe that in any case the $\sqrt{\log n}$ factor in the upper bound for thick Minkowski sums is an artifact of the proof. It would also be interesting to extend the combination theorem for unions of unit-width rectangles and unit-width discs to the case where the radius of the discs may be smaller.

References

[1] P.K. Agarwal. Range searching. In J.E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, chapter 36, pages 809–838. Chapman & Hall/CRC, Boca Raton, 2nd edition, 2004.

[2] Bernard Chazelle. Triangulating a simple polygon in linear time. *Discrete Comput. Geom.*, 6(5):485–524, 1991.

[3] Bernard Chazelle, H. Edelsbrunner, Leonidas J. Guibas, and Micha Sharir. Algorithms for bichromatic line segment problems and polyhedral terrains. *Algorithmica*, 11:116–132, 1994.

[4] K. Daniels and V. J. Milenkovic. Multiple translational containment, part i: An approximate algorithm. *Algorithmica*, 19(1–2):148–182, September 1997.

[5] M. de Berg, M.J. Katz, A.F. van der Stappen, and J. Vleugels. Realistic input models for geometric algorithms. *Algorithmica*, 34:81–97, 2002.

[6] A. Efrat. The complexity of the union of (α, β)-covered objects. In *Proc. 15th Annu. ACM Sympos. Computat. Geometry*, 1999.

[7] A. Efrat, M. J. Katz, F. Nielsen, and M. Sharir. Dynamic data structures for fat objects and their applications. *Comput. Geom. Theory Appl.*, 15:215–227, 2000.

[8] A. Efrat, G. Rote, and M. Sharir. On the union of fat wedges and separating a collection of segments by a line. *Comput. Geom. Theory Appl.*, 3:277–288, 1993.

[9] A. Efrat and M. Sharir. On the complexity of the union of fat objects in the plane. *Discrete Comput. Geom.*, 23:171–189, 2000.

[10] J. Erickson. Local polyhedra and geometric graphs. In *Proc. 19th Annu. ACM Sympos. Computat. Geometry*, pages 171–180, 2003.

[11] A. Gajentaan and M. H. Overmars. On a class of $O(n^2)$ problems in computational geometry. *Comput. Geom. Theory Appl.*, 5:165–185, 1995.

[12] D. Halperin. Arrangements. In J.E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, chapter 24, pages 529–562. Chapman & Hall/CRC, Boca Raton, 2nd edition, 2004.

[13] R. Heckmann and T. Lengauer. Computing upper and lower bounds on textile nesting problems. In *Proc. 4th Annu. European Sympos. Algorithms*, volume 1136 of *Lecture Notes Comput. Sci.*, pages 392–405. Springer-Verlag, 1996.

[14] K. Kedem, R. Livne, J. Pach, and Micha Sharir. On the union of Jordan regions and collision-free translational motion amidst polygonal obstacles. *Discrete Comput. Geom.*, 1:59–71, 1986.

[15] J.-C. Latombe. *Robot Motion Planning.* Kluwer Ac. Pub. Boston, 1991.

[16] J. Matoušek, J. Pach, Micha Sharir, S. Sifrony, and Emo Welzl. Fat triangles determine linearly many holes. *SIAM J. Comput.*, 23:154–169, 1994.

[17] Victor J. Milenkovic. Multiple translational containment, part ii: Exact algorithm. *Algorithmica*, 19(1–2):183–218, September 1997.

[18] M.H. Overmars and A.F. van der Stappen. Range searching and point location among fat objects. *Journal of Algorithms*, 21:629–656, 1996.

[19] J. Pach and M. Sharir. On the boundary of the union of planar convex sets. *Discrete Comput. Geom.*, 21:321–328, 1999.

[20] J. Pach and G. Tardos. On the boundary complexity of the union of fat triangles. *SIAM J. Comput.*, 31:1745–1760, 2002.

[21] Janos Pach and Pankaj K. Agarwal. *Combinatorial Geometry.* John Wiley & Sons, New York, NY, 1995.

[22] M. Sharir. Algorithmic motion planning. In J.E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, chapter 47, pages 1037–1064. Chapman & Hall/CRC, Boca Raton, 2nd edition, 2004.

[23] A. F. van der Stappen and M. H. Overmars. Motion planning amidst fat obstacles. In *Proc. 10th Annu. ACM Sympos. Comput. Geom.*, pages 31–40, 1994.

[24] M. van Kreveld and B. Speckmann. Cutting a country for smallest square fit. In *Proc. ISAAC'02*, volume 2518 of *Lecture Notes in Computer Science*, pages 91–102. Springer, 2002.

[25] Marc van Kreveld. On fat partitioning, fat covering, and the union size of polygons. *Comput. Geom. Theory Appl.*, 9(4):197–210, 1998.