




# Clique-Based Separators for Geometric Intersection Graphs

Mark de Berg<sup>1</sup> · Sándor Kisfaludi-Bak<sup>2</sup> · Morteza Monemizadeh<sup>1</sup> ·  
Leonidas Theocharous<sup>1</sup> 

Received: 7 November 2021 / Accepted: 26 August 2022 / Published online: 15 October 2022  
© The Author(s) 2022

## Abstract

Let  $F$  be a set of  $n$  objects in the plane and let  $\mathcal{G}^\times(F)$  be its intersection graph. A balanced clique-based separator of  $\mathcal{G}^\times(F)$  is a set  $\mathcal{S}$  consisting of cliques whose removal partitions  $\mathcal{G}^\times(F)$  into components of size at most  $\delta n$ , for some fixed constant  $\delta < 1$ . The weight of a clique-based separator is defined as  $\sum_{C \in \mathcal{S}} \log(|C| + 1)$ . Recently De Berg et al. (SIAM J. Comput. 49: 1291–1331, 2020) proved that if  $\mathcal{S}$  consists of convex fat objects, then  $\mathcal{G}^\times(F)$  admits a balanced clique-based separator of weight  $O(\sqrt{n})$ . We extend this result in several directions, obtaining the following results. (i) Map graphs admit a balanced clique-based separator of weight  $O(\sqrt{n})$ , which is tight in the worst case. (ii) Intersection graphs of pseudo-disks admit a balanced clique-based separator of weight  $O(n^{2/3} \log n)$ . If the pseudo-disks are polygonal and of total complexity  $O(n)$  then the weight of the separator improves to  $O(\sqrt{n} \log n)$ . (iii) Intersection graphs of geodesic disks inside a simple polygon admit a balanced clique-based separator of weight  $O(n^{2/3} \log n)$ . (iv) Visibility-restricted unit-disk graphs in a polygonal domain with  $r$  reflex vertices admit a balanced clique-based separator of weight  $O(\sqrt{n} + r \log(n/r))$ , which is tight in the worst case. These results immediately imply sub-exponential

---

S. Kisfaludi-Bak: The research was conducted while the author was at the Max Planck Institute for Informatics, Saarbrücken, Germany.

---

✉ Leonidas Theocharous  
l.theocharous@tue.nl

Mark de Berg  
m.t.d.berg@tue.nl

Sándor Kisfaludi-Bak  
sandor.kisfaludi-bak@aalto.fi

Morteza Monemizadeh  
m.monemizadeh@tue.nl

<sup>1</sup> Department of Computer Science, TU Eindhoven, Eindhoven, The Netherlands

<sup>2</sup> Department of Computer Science, Aalto University, Espoo, Finland

algorithms for MAXIMUM INDEPENDENT SET (and, hence, VERTEX COVER), for FEEDBACK VERTEX SET, and for  $q$ -COLORING for constant  $q$  in these graph classes.

**Keywords** Computational geometry · Intersection graphs · Separator theorems

## 1 Introduction

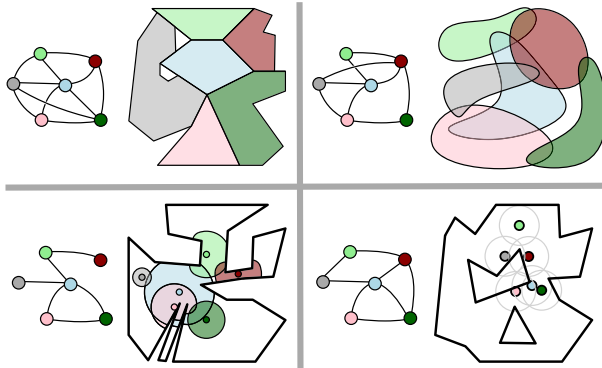
The famous Planar Separator Theorem states that any planar graph  $\mathcal{G} = (V, E)$  with  $n$  nodes<sup>1</sup> admits a subset  $\mathcal{S} \subset V$  of size  $O(\sqrt{n})$  nodes whose removal decomposes  $\mathcal{G}$  into connected components of size at most  $2n/3$ . The subset  $\mathcal{S}$  is called a balanced<sup>2</sup> *separator* of  $\mathcal{G}$ . The theorem was first proved in 1979 by Lipton and Tarjan [21], and it has been instrumental in the design of algorithms for planar graphs: it has been used to design efficient divide-and-conquer algorithms, to design sub-exponential algorithms for various NP-hard graph problems, and to design approximation algorithms for such problems.

The Planar Separator Theorem has been extended to various other graph classes. Our interest lies in *geometric intersection graphs*, where the nodes correspond to geometric objects and there is an arc between two nodes iff the corresponding objects intersect. If the objects are disks, the resulting graph is called a *disk graph*. Disk graphs, and in particular unit-disk graphs, are a popular model for wireless communication networks and have been studied extensively. Miller et al. [25] and Smith and Wormald [30] showed that if  $F$  is a set of balls in  $\mathbb{R}^d$  of ply at most  $k$ —the *ply* of  $F$  is the maximum number of objects in  $F$  with a common intersection—then the intersection graph of  $F$  has a separator of size  $O(k^{1/d} n^{1-1/d})$ . This was generalized by Chan [6] and Har-Peled and Quanrud [16] to intersection graphs of so-called low-density sets. Separators for string graphs—a string graph is an intersection graph of sets of curves in the plane—have also been considered [15, 20, 24], with Lee [20] showing that a separator of size  $O(\sqrt{m})$  exists, where  $m$  is the number of arcs of the graph.

Even for simple objects such as disks or squares, one must restrict the ply to obtain a separator of small size. Otherwise the objects can form a single clique, which obviously does not have a separator of sublinear size. To design subexponential algorithms for problems such as MAXIMUM INDEPENDENT SET, however, one can also work with a separator consisting of a small number of cliques instead of a small number of nodes. Such *clique-based separators* were introduced recently by De Berg et al. [11]. Formally, a clique-based separator of a graph  $\mathcal{G}$  is a collection  $\mathcal{S}$  of node-disjoint cliques whose union is a balanced separator of  $\mathcal{G}$ . The *weight* of  $\mathcal{S}$  is defined as  $\text{weight}(\mathcal{S}) := \sum_{C \in \mathcal{S}} \log(|C| + 1)$ . De Berg et al. [11] proved that the intersection graph of any set  $F$  of  $n$  convex fat objects in the plane admits a clique-based separator of weight  $O(\sqrt{n})$ , and they used this to obtain algorithms with running time  $2^{O(\sqrt{n})}$  for many classic NP-hard problems on such graphs. This running time is optimal, assuming the Exponential-

<sup>1</sup> We use the terms *node* and *arc* when talking about graphs, and *vertex* and *edge* for geometric objects.

<sup>2</sup> For a separator to be balanced it suffices that the components have size at most  $\delta n$  for some constant  $\delta < 1$ . When we speak of separators, we always mean balanced separators, unless stated otherwise.



**Fig. 1** A map graph, a pseudo-disk graph, a geodesic-disk graph, and a visibility restricted unit-disk graph. For the latter class, the grey disks in the picture have radius  $\frac{1}{2}$

Time Hypothesis (ETH). The result generalizes to convex fat objects in  $\mathbb{R}^d$ , where the bound on the weight of the clique-based separator becomes  $O(n^{1-1/d})$ .

The goal of our paper is to investigate whether similar results are possible for non-fat objects in the plane. Note that not all intersection graphs admit clique-based separators of small weight. String graphs, for instance, can have arbitrarily large complete bipartite graphs as induced subgraphs, in which case any balanced clique-based separator has weight  $\Omega(n)$ .

The first type of intersection graphs we consider are map graphs, which are a natural generalization of planar graphs. The other types are generalizations of disk graphs. One way to generalize disk graphs is to consider fat objects instead of disks, as done by De Berg et al. [11]. We will study three other generalizations, involving non-fat objects: pseudo-disks, geodesic disks, and visibility-restricted unit disks. Next we define the graph classes we consider more precisely; see Fig. 1 for an example of each graph class.

In the following, we use  $\mathcal{G}^\times(F)$  to denote the intersection graph induced by a set  $F$  of objects. For convenience, we do not distinguish between the objects and the corresponding nodes, so we use  $F$  to denote the set of objects as well as the set of nodes in  $\mathcal{G}^\times(F)$ . We assume that the objects in  $F$  are connected, bounded, and closed. Throughout the paper, we will use  $\log$  for the base 2 logarithm and  $\ln$  for the base  $e$  logarithm.

**Map graphs** Let  $\mathcal{M}$  be a planar subdivision and  $F$  be its set of faces. The graph with node set  $F$  that has an arc between every pair of neighboring faces is called the *dual graph* of  $\mathcal{M}$ , and it is planar. Here two faces are neighbors if their boundaries have an edge of the subdivision in common. A *map graph* [8] is defined similarly, except now two faces are neighbors even if their boundaries meet in a single point. Alternatively, we can define a map graph as the intersection graph of a set  $F$  of interior-disjoint regions in the plane. Since arbitrarily many faces can share a vertex on their boundary, map graphs can contain arbitrarily large cliques. If at most  $k$  faces meet at each subdivision vertex, the graph is called a *k-map graph*. Chen [7] proved that any

$k$ -map graph has a (normal, not clique-based) separator of size  $O(\sqrt{kn})$ , which is also implied by Lee's recent result on string graphs [20].

*Pseudo-disk graphs* A set  $F$  of objects is a set of pseudo-disks if for any  $f, f' \in F$  the boundaries  $\partial f$  and  $\partial f'$  intersect at most twice. Pseudo-disks were introduced in the context of motion planning by Kedem et al. [17], who proved that the union complexity of  $n$  pseudo-disks is  $O(n)$ . Since then they have been studied extensively. We consider two types of pseudo-disks: polygonal pseudo-disks with  $O(n)$  vertices in total, and arbitrary pseudo-disks.

*Geodesic-disk graphs and visibility-restricted unit-disk graphs* As mentioned, unit-disk graphs are popular models for wireless communication networks. We consider two natural generalizations of unit-disk graphs, which can be thought of as communication networks in a polygonal environment that may obstruct communication.

- *Geodesic-disk graphs* in a simple polygon  $P$  are intersection graphs of geodesic disks inside  $P$ . (The *geodesic disk* with center  $q \in P$  and radius  $r$  is the set of all points in  $P$  at geodesic distance at most  $r$  from  $q$ , where the geodesic distance between two points is the length of the shortest path between them inside  $P$ .)
- In *visibility-restricted unit-disk graphs* the nodes correspond to a set  $Q$  of  $n$  points inside a polygon  $P$ , which may have holes, and two points  $p, q \in Q$  are connected by an arc iff  $|pq| \leq 1$  and  $p$  and  $q$  see each other (meaning that  $pq \subset P$ ).<sup>3</sup> A more general, directed version of such graphs was studied by Ben-Moshe et al. [2] under the name range-restricted visibility graph. They presented an output-sensitive algorithm to compute the graph.

## 1.1 Our Results: Clique-Based Separator Theorems

So far, clique-based separators were studied for fat objects: De Berg et al. [11] consider convex or similarly-sized fat objects, Kisfaludi-Bak et al. [19] study how the fatness of axis-aligned fat boxes impacts the separator weight, and Kisfaludi-Bak [18] studies balls in hyperbolic space. The  $O(\sqrt{n})$  bound on the separator weight is tight even for unit-disk graphs. Indeed, a  $\sqrt{n} \times \sqrt{n}$  grid graph can be realized as a unit-disk graph, and any separator of such a grid graph must contain  $\Omega(\sqrt{n})$  nodes. Since the maximum clique size in a grid graph is two, any separator must contain  $\Omega(\sqrt{n})$  cliques. All graph classes we consider can realize a  $\sqrt{n} \times \sqrt{n}$  grid graph, so  $\Omega(\sqrt{n})$  is a lower bound on the weight of the clique-based separators we consider. We obtain the following results.

In Sect. 2 we show that any map graph has a clique-based separator of weight  $O(\sqrt{n})$ . This gives the first ETH-tight algorithms for MAXIMUM INDEPENDENT SET (and, hence, VERTEX COVER), FEEDBACK VERTEX SET, and COLORING in map graphs; see below.

In Sect. 2.1 we show that any intersection graph of pseudo-disks has a clique-based separator of weight  $O(n^{2/3} \log n)$ . If the pseudo-disks are polygonal and of total complexity  $O(n)$  then the weight of the separator improves to  $O(\sqrt{n} \log n)$ .

<sup>3</sup> Visibility-restricted unit-disk graphs are, strictly speaking, not intersection graphs. In particular, if  $R_q$  is defined as the region of points within  $P$  that are visible from  $q$  and lie within distance  $1/2$ , then the visibility-restricted unit-disk graph is *not* the same as the intersection graph of the objects  $R_q$ .

In Sect. 3 we consider intersection graphs of geodesic disks inside a simple polygon. At first sight, geodesic disks seem not much harder to deal with than fat objects: they can have skinny parts only in narrow corridors and then packing arguments may still be feasible. Unfortunately another obstacle prevents us from applying a packing argument: geodesic distances in a simply connected polygon induce a metric space whose doubling dimension depends on the number of reflex vertices of the polygon. Nevertheless, by showing that geodesic disks inside a simple polygon behave as pseudo-disks, we are able to obtain a clique-based separator of weight  $O(n^{2/3} \log n)$ , independent of the number of reflex vertices.

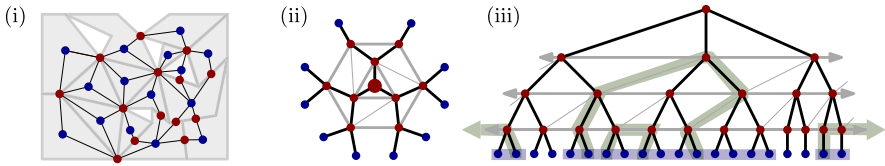
In Sect. 4 we study visibility-restricted unit-disk graphs. We give an  $\Omega(\min(n, r \log(n/r)) + \sqrt{n})$  lower bound for the separator weight, where  $r$  is the number of reflex vertices of the polygon in which the graph is defined. We thus show that a clique-based separator whose weight depends only on  $n$ , the number of points defining the visibility graph, is not possible. We then show how to construct a clique-based separator of weight  $O(\min(n, r \log(n/r)) + \sqrt{n})$ .

All separators can be computed in polynomial time. For map graphs and for the pseudo-disk intersection graphs, we assume the objects have total complexity  $O(n)$ . If the objects have curved edges, we assume that basic operations (such as computing the intersection points of two such curves) take  $O(1)$  time.

## 1.2 Applications

In Sect. 5 we apply our separator theorems to obtain subexponential algorithms for MAXIMUM INDEPENDENT SET, FEEDBACK VERTEX SET, and  $q$ -COLORING for constant  $q$  in the graph classes discussed above. The crucial property of these problems that makes our separator applicable, is that the possible ways in which a solution can “interact” with a clique of size  $k$  is polynomial in  $k$ . We use known techniques (mostly from De Berg et al. [11]) to solve the three problems on any graph class that has small clique-based separators.

All our graph classes are subsumed by string graphs. Bonnet and Rzazewski [4] showed that string graphs have  $2^{O(n^{2/3} \log n)}$  algorithms for MAXIMUM INDEPENDENT SET and 3-COLORING, and a  $2^{n^{2/3} \log^{O(1)} n}$  algorithm for FEEDBACK VERTEX SET, and that string graphs do not have subexponential algorithms for  $q$ -COLORING with  $q \geq 4$  under ETH. One can also obtain subexponential algorithms in some of our classes from results of Fomin et al. [7, 14] or Marx and Philipczuk [22]. The running times we obtain match or slightly improve the results that can be obtained from these existing results. It should be kept in mind, however, that the existing results are for more general graph classes. An exception are our results on map graphs, which were explicitly studied before and where we improve the running time for MAXIMUM INDEPENDENT SET and FEEDBACK VERTEX SET from  $2^{O(\sqrt{n} \log n)}$  to  $2^{O(\sqrt{n})}$ . (But, admittedly, the existing results apply in the parameterized setting while ours don't.) In any case, the main advantage of our approach is that it allows us to solve MAXIMUM INDEPENDENT SET, FEEDBACK VERTEX SET and  $q$ -COLORING on each of the mentioned graph classes in a uniform manner.



**Fig. 2** (i) A witness graph for the map graph induced by the grey regions. Points in  $P$  are blue, points in  $Q$  are red. (ii) The gadget used to replace a witness point. The edges of  $T_q$  are black, the cycles connecting nodes at the same level are grey and thick, the edges to triangulate the 4-cycle are grey and thin. (iii) The green paths show an example of how the separator can intersect a gadget. (Note that the tree “wraps around”, as in part (ii) of the figure; see also one of the green paths.) The objects added to the clique  $C_q$  correspond to the leaves indicated by the blue rectangles

## 2 Map Graphs

Recall that a map graph is the intersection graph of a set  $F$  of interior-disjoint objects in the plane. We construct a clique-based separator for  $\mathcal{G}^\times(F)$  in four steps. First, we construct a bipartite plane *witness graph*  $\mathcal{H}_1$  with node set  $P \cup Q$ , where the nodes in  $P$  correspond to the objects in  $F$  and the nodes in  $Q$  (with their incident arcs) model the adjacencies in  $\mathcal{G}^\times(F)$ . Next, we replace each node  $q \in Q$  by a certain gadget whose “leaves” are the neighbors of  $q$ , and we triangulate the resulting graph. We then apply the Planar Separator Theorem to obtain a separator for the resulting graph  $\mathcal{H}_2$ . Finally, we turn the separator for  $\mathcal{H}_2$  into a clique-based separator for  $\mathcal{G}^\times(F)$ . Next we explain these steps in detail.

*Step 1: Creating a witness graph* To construct a witness graph for  $\mathcal{G}^\times(F)$  we use the method of Chen et al. [8]: take a point  $p_f$  in the interior of each object  $f \in F$ , and take a *witness point*  $q \in \partial f \cap \partial f'$  for each pair of touching objects  $f, f' \in F$  and add arcs from  $q$  to the points  $p_f$  and  $p_{f'}$ .

Let  $P = \{p_f : f \in F\}$  and let  $Q$  be the set of all witness points added. We denote the resulting bipartite graph with node set  $P \cup Q$  by  $\mathcal{H}_1$ ; see Fig. 2(i) for an example. Observe that points where many objects meet can serve as witness points for many neighboring pairs in  $\mathcal{G}^\times(F)$ . Chen et al. [8, Lemma 2.3] proved that any map graph admits a witness set  $Q$  of size  $O(n)$ . If the objects in  $F$  are polygons with  $O(n)$  vertices in total then  $Q$  can be found in  $O(n)$  time (since the vertices can serve as the set  $Q$ .)

*Step 2: Replacing witness points by gadgets and triangulating* We would like to construct a separator for  $\mathcal{H}_1$  using the Planar Separator Theorem, and convert it to a clique-based separator for  $\mathcal{G}^\times(F)$ . For every witness point  $q \in Q$  in the separator for  $\mathcal{H}_1$ , the conversion would add a clique  $C_q$  to the clique-based separator, namely, the clique corresponding to all objects  $f \in F$  such that  $p_f$  is adjacent to  $q$ . However, the node  $q$  adds 1 to the separator size, but the clique  $C_q$  adds  $\log(|C_q| + 1)$  to the weight of the clique-based separator. To deal with this we modify  $\mathcal{H}_1$ , as follows.

Consider a node  $q \in Q$ . Let  $N(q) \subseteq P$  denote the set of neighbors of  $q$ . For all nodes  $q \in Q$  with  $|N(q)| \geq 3$ , we replace the star induced by  $\{q\} \cup N(q)$  by a gadget  $G_q$ , which is illustrated in Fig. 2(ii) and defined as follows.

First, we create a tree  $\mathcal{T}_q$  with root  $q$  and whose leaves are the nodes in  $N(q)$ , as follows. Define the *level*  $\ell(v)$  of a node  $v$  in  $\mathcal{T}_q$  to be the distance of  $v$  to the root; thus the root has level 0, its children have level 1, and so on. All leaves in  $\mathcal{T}_q$  are at the same level, denoted  $\ell_{\max}$ . The root has degree 3, nodes at level  $\ell$  with  $1 \leq \ell < \ell_{\max} - 1$  have degree 2, and nodes at level  $\ell_{\max} - 1$  have degree 2 or 1. For each  $\ell < \ell_{\max}$  we connect the nodes at level  $\ell$  into a cycle. After doing so, all faces in the gadget (except the outer face) are triangles or 4-cycles. We finish the construction by adding a diagonal in each 4-cycle. Define the *height* of a node  $v$  as  $\text{height}(v) := \ell_{\max} - \ell(v)$ . The following observation follows from the construction.

**Observation 1** *Let  $v$  be a node at height  $h > 0$  in the gadget  $G_q$ .*

- (i) *The subtree of  $\mathcal{T}_q$  rooted at  $v$ , denoted  $\mathcal{T}_q(v)$ , has at most  $3 \cdot 2^{h-1}$  leaves.*
- (ii) *The distance from  $v$  to any leaf in  $\mathcal{T}_q$  is at least  $h$ .*

To unify the exposition, it will be convenient to also create a gadget for the case where  $q$  has only two neighbors in  $\mathcal{H}_1$ , say  $p_f$  and  $p_{f'}$ . We then define  $\mathcal{T}_q$  to consist of the arcs  $(q, p_f)$  and  $(q, p_{f'})$ . Note that Observation 1 holds for this gadget as well.

By replacing each witness point  $q \in Q$  with a gadget  $G_q$  as above, we obtain a (still planar) graph. We triangulate this graph to obtain a maximal planar graph  $\mathcal{H}_2$ .

*Step 3: Constructing a separator for  $\mathcal{H}_2$*  We now want to apply the Planar Separator Theorem to  $\mathcal{H}_2$ . Our final goal is to obtain a balanced clique-based separator for  $\mathcal{G}^\times(F)$ . Hence, we want the separator for  $\mathcal{H}_2$  to be balanced with respect to  $P$ . We will also need the separator for  $\mathcal{H}_2$  to be connected. Both properties are guaranteed by the following version of the Planar Separator Theorem, which was proved by Djidjev and Venkatesan [13].

**Planar Separator Theorem.** Let  $\mathcal{G} = (V, E)$  be a maximal planar graph with  $n$  nodes. Let each node  $v \in V$  have a non-negative cost, denoted  $\text{cost}(v)$ , with  $\sum_{v \in V} \text{cost}(v) = 1$ . Then  $V$  can be partitioned in  $O(n)$  time into three sets  $A, B, S$  such that (i)  $S$  is a simple cycle of size  $O(\sqrt{n})$ , (ii)  $\mathcal{G}$  has no arcs between a node in  $A$  and a node in  $B$ , and (iii)  $\sum_{v \in A} \text{cost}(v) \leq 2/3$  and  $\sum_{v \in B} \text{cost}(v) \leq 2/3$ .

When applying the Planar Separator Theorem to  $\mathcal{H}_2$ , we set  $\text{cost}(p) := 1/n$  for all nodes  $p_f \in P$  and  $\text{cost}(v) := 0$  for all other nodes. We denote the resulting separator for  $\mathcal{H}_2$  by  $\mathcal{S}(\mathcal{H}_2)$  and the node sets inside and outside the separator by  $A(\mathcal{H}_2)$  and  $B(\mathcal{H}_2)$ , respectively.

*Step 4: Turning the separator for  $\mathcal{H}_2$  into a clique-based separator for  $\mathcal{G}^\times(F)$*  We convert  $\mathcal{S}(\mathcal{H}_2)$  into a clique-based separator  $\mathcal{S}$  for  $\mathcal{G}^\times(F)$  as follows.

- For each node  $p_f \in \mathcal{S}(\mathcal{H}_2) \cap P$  we put the (singleton) clique  $\{f\}$  into  $\mathcal{S}$ .
- For each gadget  $G_q$  we proceed as follows. Let  $V_q$  be the set of all nodes  $v \in \mathcal{T}_q$  that are in  $\mathcal{S}(\mathcal{H}_2)$ , and define  $C_q := \{f \in F : p_f \text{ is a leaf of } \mathcal{T}_q(v) \text{ that has an ancestor in } V_q\}$ ; see Fig. 2(iii). Observe that  $C_q$  is a clique in  $\mathcal{G}^\times(F)$ . We add<sup>4</sup>  $C_q$  to  $\mathcal{S}$ .

<sup>4</sup> We tacitly assume that if an object is in multiple cliques in  $\mathcal{S}$ , we remove all but one of its occurrences.



The clique-based separator  $\mathcal{S}$  induces a partition of  $F \setminus \bigcup_{C \in \mathcal{S}} C$  into two parts  $A$  and  $B$ , with  $|A|, |B| \leq 2n/3$ , in a natural way, namely as  $A := \{f \in F : f \notin \bigcup_{C \in \mathcal{S}} C \text{ and } p_f \in A(\mathcal{H}_2)\}$  and  $B := \{f \in F : f \notin \bigcup_{C \in \mathcal{S}} C \text{ and } p_f \in B(\mathcal{H}_2)\}$ . The following lemma ensures that  $\mathcal{S}$  is a valid separator.

**Lemma 2** *There are no arcs in  $\mathcal{G}^\times(F)$  between a node in  $A$  and a node in  $B$ .*

**Proof** Suppose for a contradiction that there are objects  $f \in A$  and  $f' \in B$  such that  $(f, f')$  is an arc in  $\mathcal{G}^\times(F)$ . Let  $q \in Q$  be a witness point for the arc  $(f, f')$ ; thus  $(p_f, q)$  and  $(p_{f'}, q)$  are arcs in  $\mathcal{H}_1$ . Consider the gadget  $G_q$  and the tree  $\mathcal{T}_q$ . Let  $\pi$  denote the path from  $p_f$  to  $p_{f'}$  in  $\mathcal{T}_q$ . Note that none of the nodes on  $\pi$  can be in  $\mathcal{S}(\mathcal{H}_2)$ , otherwise  $V_q$  contains an ancestor of  $p_f$  or of  $p_{f'}$ , and (at least) one of the nodes  $p_f, p_{f'}$  is in a clique that was added to  $\mathcal{S}$ . But then the nodes  $p_f, p_{f'}$  are still connected in  $\mathcal{H}_2$  after the removal of  $\mathcal{S}(\mathcal{H}_2)$ . Hence, we have  $p_f, p_{f'} \in A(\mathcal{H}_2)$  or  $p_f, p_{f'} \in B(\mathcal{H}_2)$ , both contradicting that  $f \in A$  and  $f' \in B$ .

It remains to prove that  $\mathcal{S}$  has the desired weight.

**Lemma 3** *The total weight of the separator  $\mathcal{S}$  satisfies  $\sum_{C \in \mathcal{S}} \log(|C| + 1) = O(\sqrt{n})$ .*

**Proof** Since  $\mathcal{S}(\mathcal{H}_2)$  contains  $O(\sqrt{n})$  nodes, it suffices to bound the total weight of the cliques added for the gadgets  $G_q$ . Consider a gadget  $G_q$ . Recall that  $V_q$  is the set of all nodes  $v \in \mathcal{T}_q$  that are in  $\mathcal{S}(\mathcal{H}_2)$ . We claim that  $\log(|C_q| + 1) = O(|V_q|)$ , which implies that  $\sum_q \log(|C_q| + 1) = \sum_q O(|V_q|) = O(\sqrt{n})$ , as desired. It remains to prove the claim.

Since  $\mathcal{S}(\mathcal{H}_2)$  is a simple cycle, its intersection with  $G_q$  consists of one or more paths. Each path  $\pi$  enters and exits  $G_q$  at a node in  $N(q)$ . Let  $D_\pi$  denote the set of all descendants of the nodes in  $\pi$ . We will prove that  $\log(|D_\pi| + 1) = O(|\pi|)$ , where  $|\pi|$  denotes the number of nodes of  $\pi$ . This implies the claim since  $\log(|C_q| + 1) \leq \sum_\pi \log(|D_\pi| + 1) = \sum_\pi O(|\pi|) = O(|V_q|)$ .

To prove that  $\log(|D_\pi| + 1) = O(|\pi|)$ , let  $h_{\max}$  be the maximum height of any node in  $\pi$ . Thus  $|\pi| \geq h_{\max}$  by Observation 1(ii). Consider all subtrees of height  $h_{\max}$  in  $\mathcal{T}_q$ . If  $\pi$  visits  $t$  such subtrees, then  $|\pi| \geq t$ . Moreover,  $|D_\pi| \leq 3t \cdot 2^{h_{\max}-1}$  by Observation 1(i). Hence,  $\log(|D_\pi| + 1) \leq \log(3t \cdot 2^{h_{\max}-1} + 1) < h_{\max} + \log(3t) = O(\max(h_{\max}, t)) = O(|\pi|)$ .

By putting everything together we obtain the following theorem.

**Theorem 4** *Let  $F$  be a set of  $n$  interior-disjoint regions in the plane. Then the intersection graph  $\mathcal{G}^\times(F)$  has a clique-based balanced separator of weight  $O(\sqrt{n})$ . The separator can be computed in  $O(n)$  time, assuming that the total complexity of the objects in  $F$  is  $O(n)$ .*

### 2.1 Pseudo-Disk Graphs

Our clique-based separator construction for a set  $F$  of pseudo-disks uses so-called planar supports, defined as follows. Let  $\mathcal{H}$  be a hypergraph with node set  $Q$  and hyperedge set  $H$ . A graph  $\mathcal{G}_{\text{sup}}$  is a *planar support* [29] for  $\mathcal{H}$  if  $\mathcal{G}_{\text{sup}}$  is a planar graph



with node set  $Q$  such that for any hyperedge  $h \in H$  the subgraph of  $\mathcal{G}_{\text{sup}}$  induced by the nodes in  $h$  is connected. In our application we let the node set  $Q$  correspond to a set of points stabbing all pairwise intersections between the pseudo-disks, that is, for each intersecting pair  $f, f' \in F$  there will be a point  $q \in Q$  that lies in  $f \cap f'$ . The goal is to keep the size of  $Q$  small, by capturing all intersecting pairs with few points. The hyperedges are defined by the regions in  $F$ , that is, for every  $f \in F$  there is a hyperedge  $h_f := Q \cap f$ . Let  $\mathcal{H}_Q(F)$  denote the resulting hypergraph.

**Lemma 5** *Let  $F$  be a set of  $n$  objects in the plane, let  $Q$  be a set of points stabbing all pairwise intersections in  $F$ , and let  $\mathcal{H}_Q(F)$  denote the hypergraph as defined above. If  $\mathcal{H}_Q(F)$  has a planar support  $\mathcal{G}_{\text{sup}}$  then  $\mathcal{G}^\times(F)$  has a clique-based separator of size  $O(\sqrt{|Q|})$  and weight  $O(\sqrt{|Q|} \log n)$ .*

**Proof** Let  $\mathcal{S}(\mathcal{G}_{\text{sup}})$  be a separator for  $\mathcal{G}_{\text{sup}}$  of size  $O(\sqrt{|Q|})$ , which exists by the Planar Separator Theorem, and let  $A(\mathcal{G}_{\text{sup}})$  and  $B(\mathcal{G}_{\text{sup}})$  be the corresponding separated parts. To ensure an appropriately balanced separator we use the cost-balanced version of the Planar Separator Theorem, as stated in the previous section. For each object  $f \in F$  we give one point  $q_f \in Q \cap f$  a cost of  $1/n$  and all other points cost 0. We call  $q_f$  the *representative* of  $f$ . (We assume for simplicity that each  $f \in F$  intersects at least one other object  $f' \in F$ , so we can always find a representative. Objects  $f \in F$  not intersecting any other object are singletons in  $\mathcal{G}^\times(F)$  and can be ignored.) For a point  $q \in Q$ , define  $C_q$  to be the clique in  $\mathcal{G}^\times(F)$  consisting of all objects  $f \in F$  that contain  $q$ . Our clique-based separator  $\mathcal{S}$  for  $\mathcal{G}^\times(F)$  is now defined as  $\mathcal{S} := \{C_q : q \in \mathcal{S}(\mathcal{G}_{\text{sup}})\}$ , and the two separated parts are defined as:  $A := \{f \in F : f \notin \mathcal{S} \text{ and } q_f \in A(\mathcal{G}_{\text{sup}})\}$  and  $B := \{f \in F : f \notin \mathcal{S} \text{ and } q_f \in B(\mathcal{G}_{\text{sup}})\}$ . Clearly, the size of  $\mathcal{S}$  is  $O(\sqrt{|Q|})$  and its weight is  $O(\sqrt{|Q|} \log n)$ . Moreover,  $|A|, |B| \leq 2n/3$  because  $\mathcal{S}(\mathcal{G}^*)$  is balanced with respect to the node costs.

We claim there are no arcs in  $\mathcal{G}^\times(F)$  between a node in  $A$  and a node in  $B$ . Suppose for a contradiction that there are intersecting objects  $f, f'$  such that  $f \in A$  and  $f' \in B$ . By definition of  $Q$  there is a point  $q \in Q$  that lies in  $f \cap f'$ . By the planar-support property, the hyperedge  $h_f$  induces a connected subgraph of  $\mathcal{G}_{\text{sup}}$ , so there is a path  $\pi$  that connects  $q$  to the representative  $q_f$  and such that all nodes of  $\pi$  are points in  $f \cap Q$ . No node on the path  $\pi$  can be in  $\mathcal{S}(\mathcal{G}_{\text{sup}})$ , otherwise  $f$  is in a clique that was added to  $\mathcal{S}$ . Similarly, there is a path  $\pi'$  connecting  $q_{f'}$  to  $q$  such that no point on  $\pi'$  is in  $\mathcal{S}(\mathcal{G}_{\text{sup}})$ . But then there is a path from  $q_f$  to  $q_{f'}$  in  $\mathcal{G}_{\text{sup}}$  after the removal of  $\mathcal{S}(\mathcal{G}_{\text{sup}})$ . Hence,  $q_f$  and  $q_{f'}$  are in the same part of the partition, which contradicts that  $f$  and  $f'$  are in different parts.

We conclude that  $\mathcal{S}$  is a clique-based separator with the desired properties.

**Remark 6** The witness set  $Q$  in the previous section stabs all pairwise intersections of objects in the map graph, and so  $P \cup Q$  stabs all pairwise intersections as well.  $P \cap Q$  has planar support, so we can get a separator for map graphs using Lemma 5. Its weight would be  $O(\sqrt{n} \log n)$ , however, while in the previous section we managed to get  $O(\sqrt{n})$  weight.

## 2.2 Polygonal Pseudo-Disks

We now apply Lemma 5 to obtain a clique-based separator for a set  $F$  of polygonal pseudo-disks. To this end, let  $Q$  be the set of vertices of the pseudo-disks in  $F$ . Observe that whenever two pseudo-disks intersect, one must have a vertex inside the other. Indeed, either one pseudo-disk is entirely inside the other, or an edge  $e$  of  $f$  intersects an edge  $e'$  of  $f'$ . In the latter case, one of the two edges ends inside the other pseudo-disk, otherwise there are three intersections between the boundaries. Furthermore, pseudo-disks have the *non-piercing* property:  $f \setminus f'$  is connected for any two pseudo-disks  $f, f'$ . Raman and Ray [29] proved<sup>5</sup> that the hypergraph  $\mathcal{H}_Q(F)$  of a set of non-piercing regions has a planar support for any set  $Q$ , so in particular for the set  $Q$  just defined. We can thus apply Lemma 5 to compute a clique-based separator for  $\mathcal{G}^\times(F)$ . The time to compute the separator is dominated by the computation of the planar support, which takes  $O(n^3)$  time [29].

**Theorem 7** *Let  $F$  be a set of  $n$  polygonal pseudo-disks in the plane with  $O(n)$  vertices in total. Then the intersection graph  $\mathcal{G}^\times(F)$  has a clique-based balanced separator of size  $O(\sqrt{n})$  and weight  $O(\sqrt{n} \log n)$ , which can be found in  $O(n^3)$  time.*

## 2.3 Arbitrary Pseudo-Disks

To construct a clique-based separator using Lemma 5 we need a small point set  $Q$  that stabs all pairwise intersections. Unfortunately, for general pseudo-disks a linear-size set  $Q$  that stabs all intersections need not exist: there is a collection of  $n$  disks such that stabbing all pairwise intersections requires  $\Omega(n^{4/3})$  points. (Such a collection can be derived from a construction with  $n$  lines and  $n$  points with  $\Omega(n^{4/3})$  incidences [26].) Hence, we need some more work before we can apply Lemma 5.

Our separator result for arbitrary pseudo-disks works in a more general setting, namely for sets from a family  $\mathcal{F}$  with linear union complexity. (We say that  $\mathcal{F}$  has union complexity  $U(n)$  if, for any  $n \geq 1$  and any subset  $F \subset \mathcal{F}$  of size  $n$ , the union complexity of  $F$  is  $U(n)$ .) Recall that the union complexity of a family of pseudo-disks is  $O(n)$  [17]. The next theorem states that such sets admit a clique-based separator of sublinear weight. Note that the bound only depends on the number of objects, not on their complexity.

**Theorem 8** *Let  $F$  be a set of  $n$  objects from a family  $\mathcal{F}$  of union complexity  $U(n)$ , where  $U(n) \geq n$ . Then  $\mathcal{G}^\times(F)$  has a clique-based separator of size  $O((U(n))^{2/3})$  and weight  $O((U(n))^{2/3} \log n)$ . In particular, if  $F$  is a set of pseudo-disks then  $\mathcal{G}^\times(F)$  has a clique-based separator of size  $O(n^{2/3})$  and weight  $O(n^{2/3} \log n)$ . The separator can be computed in  $O(n^3)$  time, assuming the total complexity of the objects is  $O(n)$ .*

**Proof** We construct the separator  $\mathcal{S}$  in two steps.

<sup>5</sup> Raman and Ray assume the sets  $F$  and  $Q$  defining the hypergraph are in general position. Therefore we first slightly perturb the pseudo-disks in  $F$  to get them into general position (while keeping the same intersection graph), then we take  $Q$  to be a point set coinciding with the vertex set of  $F$ , and then we slightly move the points in  $Q$  such that the hypergraph remains the same.

The first step proceeds as follows. For a point  $p$  in the plane, let  $C_p$  denote the set of objects from the (current) set  $F$  containing  $p$ . As long as there is a point  $p$  such that  $|C_p| > n^{1/3}$ , we remove  $C_p$  from  $F$  and put  $C_p$  into  $\mathcal{S}$ ; here  $n$  refers to the size of the initial set  $F$ . Thus the first step adds  $O(n^{2/3})$  cliques to  $\mathcal{S}$  with total weight  $O(n^{2/3} \log n)$ . This step can easily be implemented in  $O(n^3)$  time.

In the second step, let  $F^* \subseteq F$  denote the set of  $n^*$  remaining objects after the first step. Recall that the *ply* of a family of objects is defined as the maximum number of objects from the family with a common intersection. Then for  $F^*$  we have that  $n^* \leq n$  and  $k \leq n^{1/3}$ , where  $k$  is the ply of  $F^*$ . Let  $\mathcal{A}(F^*)$  denote the arrangement induced by  $F^*$ . Since  $F^*$  has ply  $k$ , the Clarkson-Shor technique [9] implies that the complexity of the arrangement  $\mathcal{A}(F^*)$  is  $O(k^2 \cdot U(n^*/k))$ . We can compute this arrangement in  $O(k^2 \cdot U(n^*/k) \log n) = O(n^2 \log n)$  time [12]. Take a point  $q$  in each face of the arrangement, and let  $\mathcal{Q}$  be the resulting set of  $O(k^2 \cdot U(n^*/k))$  points. The set  $\mathcal{Q}$  stabs all pairwise intersections and the dual graph  $\mathcal{G}^*$  of the arrangement  $\mathcal{A}(F^*)$  is a planar support for the hypergraph  $\mathcal{H}_{\mathcal{Q}}(F^*)$ . Hence, by Lemma 5 there is a clique-based separator  $\mathcal{S}^*$  for  $\mathcal{G}^*(F^*)$  of size  $O(k\sqrt{U(n^*/k)})$  and weight  $O(k\sqrt{U(n^*/k)} \log n^*)$ . Note that  $U(n)$  is a superadditive function [1] which implies that  $U(n/k) \leq U(n)/k$  and therefore  $k\sqrt{U(n^*/k)} \leq \sqrt{k} U(n) \leq (U(n))^{2/3}$ . By adding  $\mathcal{S}^*$  to the set  $\mathcal{S}$  of cliques generated in the first step, we obtain a clique-based separator with the desired properties.

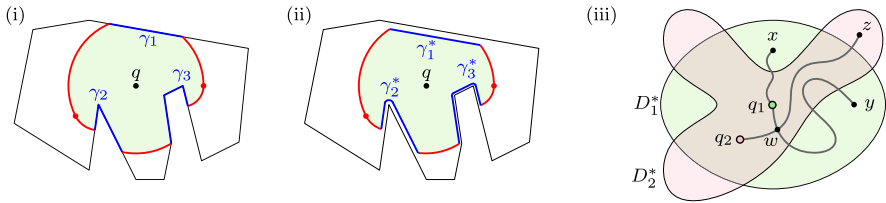
### 3 Geodesic Disks Inside a Simple Polygon

Let  $P$  be a simple polygon. We denote the shortest path (or: *geodesic*) in  $P$  between two points  $p, q \in P$  by  $\pi(p, q)$ ; note that  $\pi(p, q)$  is unique since  $P$  is simple. The *geodesic distance* between  $p$  and  $q$  is defined to be  $\|\pi(p, q)\|$ , where  $\|\pi\|$  denotes the Euclidean length of a path  $\pi$ . For a given point  $q \in P$  and radius  $r > 0$ , we call the region  $D(q, r) := \{p \in P : \|\pi(p, q)\| \leq r\}$  a *geodesic disk*. Let  $\mathcal{D} = \{D_1, \dots, D_n\}$  be a set of geodesic disks in  $P$ . To construct a clique-based separator for  $\mathcal{G}^\times(\mathcal{D})$  we will show that  $\mathcal{D}$  behaves as a set of pseudo-disks so we can apply the result of the previous section.

#### 3.1 The Structure of a Geodesic Disk

The boundary  $\partial D(q, r)$  of a geodesic disk  $D(q, r)$  consists of circular arcs lying in the interior of  $P$  (centered at  $q$  or at a reflex vertex of  $P$ ) and parts of the edges of  $P$ . We split  $\partial D(q, r)$  into *boundary pieces* at the points where the circular arcs meet  $\partial P$ . This generates two sets of boundary pieces: a set containing the pieces that consist of circular arcs, and a set  $\Gamma(D)$  containing the pieces that consist of parts of edges of  $P$ . An example can be seen in Fig. 3.

A region  $R \subseteq P$  is *geodesically convex* if for any points  $p, q \in R$  we have  $\pi(p, q) \subseteq R$ . Pollack et al. [28] showed that geodesic disks inside a simple polygon are geodesically convex. An immediate consequence is that the intersection of two geodesic disks is connected.



**Fig. 3** (i) A geodesic disk  $D$  with center  $q$  and radius  $r$ . The set  $\Gamma(D)$  has three pieces,  $\gamma_1, \gamma_2$  and  $\gamma_3$ , shown in blue. (ii) The result of the perturbation. Note that  $|\gamma_1| < |\gamma_2| < |\gamma_3|$  and so  $\varepsilon_{\gamma_1} > \varepsilon_{\gamma_2} > \varepsilon_{\gamma_3}$ . (iii) Illustration for the proof of Theorem 11 (Color figure online)

### 3.2 Geodesic Disks Behave as Pseudo-Disks

Geodesic disks in a simple polygon are not proper pseudo-disks. For example, if  $D_1$  and  $D_2$  are the blue and pink pseudo-disk in the third image in Fig. 1, then  $D_1 \setminus D_2$  has two components, which is not allowed for pseudo-disks. Nevertheless, we will show that  $\mathcal{D}$  behaves as a set of pseudo-disks in the sense that a small perturbation turns them into pseudo-disks, while keeping the intersection graph the same.

As a first step in the perturbation, we increase the radius of each geodesic disk  $D_i \in \mathcal{D}$  by some small  $\varepsilon_i$ . We pick these  $\varepsilon_i$  such that the intersection graph  $\mathcal{G}^\times(\mathcal{D})$  stays the same while all degeneracies disappear. In particular, the boundary pieces of different geodesic disks have different lengths after this perturbation, and no two geodesic disks touch. With a slight abuse of notation, we still denote the resulting set of geodesic disks by  $\mathcal{D}$ .

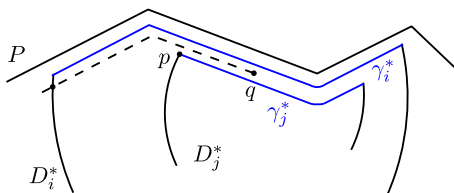
The second step in the perturbation moves each  $\gamma \in \cup_{i=1}^n \Gamma(D_i)$  into the interior of the polygon over some distance  $\varepsilon_\gamma$ , which is smaller than any of the perturbation distances chosen in the first step. More formally, for each  $\gamma \in \Gamma(D_i)$  we remove all points from  $D_i$  that are at distance less than  $\varepsilon_\gamma$  from  $\gamma$ ; see Fig. 3(ii). To ensure this gives a set of pseudo-disks we choose the perturbation distances  $\varepsilon_\gamma$  according to the reverse order of the Euclidean lengths of the pieces. That is, if  $\|\gamma\| > \|\gamma'\|$  then we pick  $\varepsilon_\gamma < \varepsilon_{\gamma'}$ . The crucial property of this scheme is that whenever  $\gamma_i \subset \gamma_j$  then  $\gamma_i$  is moved more than  $\gamma_j$ .

We denote the perturbed version of  $D_i$  by  $D_i^*$  and define  $\mathcal{D}^* := \{D_i^* : D_i \in \mathcal{D}\}$ .

**Remark 9** As already stated, the intersection of two geodesic disks is connected. This property is not invalidated by the perturbation. To see this, consider the intersection  $D_1 \cap D_2$  of two geodesic disks before the perturbation. The only way that  $D_1^* \cap D_2^*$  would not be connected, is if the perturbation transforms  $\partial(D_1 \cap D_2)$  into a curve that intersects itself. Recall that we are only actually perturbing parts of the boundary of a geodesic disk that are also parts of the boundary of  $P$ . So if we think of the perturbation as a “shrinking” of  $P$ , we need to ensure that this does not make the polygon self-intersect. For this it suffices to find the minimum over all distances between vertices of  $P$  and their non-incident edges of  $P$ . Let  $\varepsilon$  denote this distance. Then it suffices to choose all the perturbation distances smaller than  $\varepsilon/2$ .

The perturbed versions have the following important property:

**Fig. 4** Illustration for proof of Lemma 10



**Lemma 10** Let  $D_i^*, D_j^* \in \mathcal{D}^*$ . Every connected component of  $D_i^* \setminus D_j^*$  contains a point  $u$  with  $u \in D_i \setminus D_j$ .

**Proof** The only case of interest is when there exist  $\gamma_i \in \Gamma(D_i)$  and  $\gamma_j \in \Gamma(D_j)$  such that  $\gamma_i \cap \gamma_j \neq \emptyset$ . Suppose now there is a connected component of  $D_i^* \setminus D_j^*$  which does not contain such a point. Since  $D_i^* \subset D_i$  and  $D_j^* \subset D_j$ , this connected component must have been introduced due to the perturbation (in other words its intersection with all connected components of  $D_i \setminus D_j$  must be empty). This means that this connected component must be the region contained between  $\gamma_i^*$  and  $\gamma_j^*$  and which was previously contained in  $D_i \cap D_j$ . It is enough to only examine the case when  $\gamma_i$  is bigger than  $\gamma_j$  as otherwise this component only contains points of  $D_j^*$ . There are two cases:  $\gamma_j \subset \gamma_i$ , or  $\gamma_j \not\subset \gamma_i$ . In both cases note that there will exist an endpoint  $p$  of  $\gamma_j^*$  that lies inside  $D_i^*$ . Choose now a point  $q$  in  $(D_j \setminus D_j^*) \setminus (D_i \setminus D_i^*)$  at distance  $\varepsilon$  from  $\gamma_j$  with  $\varepsilon_{\gamma_i} < \varepsilon < \varepsilon_{\gamma_j}$ . See Fig. 4 for an illustration of the case  $\gamma_j \subset \gamma_i$ . Consider the dashed path starting at  $q$  that stays at distance  $\varepsilon$  from  $\partial P$  going towards  $p$ . Due to the definition of our perturbation this path will stay inside  $D_i^*$  and remain outside  $D_j^*$ . Hence, after exiting  $D_j^*$  near the point  $p$  we reach a point  $u \in D_i \setminus D_j$ , which is a contradiction.

**Theorem 11** Any set  $\mathcal{D}$  of geodesic disks inside a simple polygon  $P$  can be slightly perturbed such that the resulting set  $\mathcal{D}^*$  is a set of pseudo-disks with  $\mathcal{G}^\times(\mathcal{D}) = \mathcal{G}^\times(\mathcal{D}^*)$ .

**Proof** Consider the set  $\mathcal{D}^*$  resulting from the perturbation described above. Suppose for a contradiction that there exist two objects  $D_1^*, D_2^* \in \mathcal{D}^*$  such that  $\partial D_1^*$  and  $\partial D_2^*$  cross four or more times. By Remark 9 we know that  $D_1^* \cap D_2^*$  is connected. Hence, if  $\partial D_1^*$  and  $\partial D_2^*$  cross four or more times then  $D_1^* \setminus D_2^*$  (and, similarly,  $D_2^* \setminus D_1^*$ ) has two or more components.

For  $i = 1, 2$ , let  $q_i$  and  $r_i$  denote the center and radius of  $D_i$ . Without loss of generality assume that  $r_1 \leq r_2$ . Let  $x$  and  $y$  be points in different components of  $D_1^* \setminus D_2^*$ ; see Fig. 3 (iii). By Lemma 10 we can pick  $x$  and  $y$  such that  $x, y \in D_1 \setminus D_2$ . By concatenating the geodesics  $\pi(x, q_1)$  and  $\pi(q_1, y)$  we obtain a curve that splits  $D_2^*$  into at least two parts—this is independent of where  $q_1$  lies, or whether  $\pi(x, q_1)$  and  $\pi(q_1, y)$  partially overlap. (Note that these geodesics lie in  $D_1$  but not necessarily in  $D_1^*$ . However, they cannot “go around” a component of  $D_2^* \setminus D_1^*$ , because  $D_1$  cannot fully contain such a component by Lemma 10. Hence,  $\pi(x, q_1) \cup \pi(q_1, y)$  must indeed go through  $D_2^*$ .) Not all components of  $D_2^* \setminus D_1^*$  can belong to the same part, otherwise  $x$  and  $y$  would not be in different components of  $D_1^* \setminus D_2^*$ . Take a point  $z \in D_2^* \setminus D_1^*$

that lies in a different part than  $q_2$ , the center of  $D_2$ . Again by Lemma 10 we can pick  $z$  such that  $z \in D_2 \setminus D_1$ . Then the geodesic  $\pi(q_2, z)$  must cross  $\pi(x, q_1) \cup \pi(q_1, y)$ , say at a point  $w \in \pi(q_1, y)$ . Since  $z \notin D_1$  and  $y \notin D_2$  we must have  $\|\pi(q_1, w) \cup \pi(w, z)\| + \|\pi(q_2, w) \cup \pi(w, y)\| > r_1 + r_2$ . But this gives a contradiction because  $y \in D_1$  and  $z \in D_2$  implies  $\|\pi(q_1, w) \cup \pi(w, y)\| + \|\pi(q_2, w) \cup \pi(w, z)\| \leq r_1 + r_2$ .

It remains to show that  $\mathcal{G}^\times(\mathcal{D}) = \mathcal{G}^\times(\mathcal{D}^*)$ . As mentioned earlier, the increase of the radii in the first step of the perturbation is chosen sufficiently small so that no new intersections are introduced. The second step shrinks the geodesic disks, so no new intersections are introduced in that step either. Finally, the fact that the perturbations in the second step are smaller than in the first step guarantees that no intersections are removed.

Theorem 11 allows us to apply Theorem 8. When doing so, we actually do not need to perturb the geodesic disks. We only use the perturbation to argue that the number of faces in the arrangement defined by  $n$  geodesic disks of ply  $k$  is  $O(nk)$ . Computing the geodesic disks (and then computing the separator) can be done in polynomial time in  $n$  and the number of vertices of  $P$ . We obtain the following result.

**Corollary 12** *Let  $\mathcal{D}$  be a set of  $n$  geodesic disks inside a simple polygon with  $m$  vertices. Then  $\mathcal{G}^\times(\mathcal{D})$  has a clique-based separator of size  $O(n^{2/3})$  and weight  $O(n^{2/3} \log n)$ , which can be computed in time polynomial in  $n$  and  $m$ .*

## 4 Visibility-Restricted Unit-Disk Graphs Inside a Polygon

Let  $P$  be a simple polygon, possibly with holes, and let  $Q$  be a set of  $n$  points inside  $P$ . We define  $\mathcal{G}_{\text{vis}, P}^\times(Q)$  to be the visibility-restricted unit-disk graph of  $Q$ . The nodes in  $\mathcal{G}_{\text{vis}, P}^\times(Q)$  correspond to the points in  $Q$  and there is an edge between two points  $p, q \in Q$  iff  $|pq| \leq 1$  and  $p$  and  $q$  see each other. A vertex of  $P$  is *reflex* if its angle within the polygon is more than 180 degrees; note that for a vertex of a hole we look at the angle within  $P$ , not within the hole. Our separator construction will make use of the notion of a *centerpoint* of  $P$  which we introduce below.

### 4.1 Centerpoint of a Simple Polygon

This is a natural extension of the notion of the centerpoint for a set  $Q$  of  $n$  points in the plane, which is a point  $p$  (not necessarily from  $Q$ ) such that any line through  $p$  divides the plane in two half-planes each containing at most  $2n/3$  of the points from  $Q$ . Recall that a *chord* in  $P$  is a line segment  $s \subset P$  that connects two points on the boundary of  $P$ . Since we consider  $P$  to be a closed set, a chord may pass through one or more reflex vertices of  $P$ . We say that a chord is *maximal* if it cannot be extended without exiting the polygon. We define a *half-polygon* of  $P$  to be a sub-polygon of  $P$  that is bounded by a (not necessarily maximal) chord  $s$  and a portion  $\gamma \subset \partial P$  of the boundary of  $P$  such that  $s \cap \gamma = \{z_1, z_2\}$ , where  $z_1, z_2$  are the endpoints of  $s$ . Note that any chord splits  $P$  into two or more half-polygons.

**Definition 13** (Centerpoint in a Simple Polygon) Let  $P$  be a simple polygon and  $Q$  be a set of  $n$  points in  $P$ . A *centerpoint for  $Q$  in  $P$*  is a point  $p \in P$  such that any maximal chord through  $p$  splits  $P$  into half-polygons that each contain at most  $2n/3$  points from  $Q$  in their interior.

**Theorem 14** For any simple polygon  $P$  and point set  $Q \subset P$ , a centerpoint for  $Q$  in  $P$  exists.

**Proof** To prove the theorem we can follow the standard proof [23] of the existence of a centerpoint in the plane almost verbatim. Thus, consider the family  $H$  of all half-polygons of  $P$  whose interior contains more than  $2n/3$  points of  $Q$ . Any three of these half-polygons must intersect, because the complement of each of them contains less than  $n/3$  points from  $Q$ . Recall that Helly's Theorem states that if we have a family of convex sets in the plane such that any three of them intersect, then they all intersect in a common point. While our half-polygons are not convex, the same result is true here, as follows from Molnar's Theorem [5] which states that if a family of simply connected compact sets in the plane is such that any two members have a connected intersection and any three members have a non-empty intersection, then the intersection of the family is non-empty. Our family  $H$  satisfies these conditions and hence  $\bigcap H \neq \emptyset$ .

We claim that any point  $c$  in the common intersection of the half-polygons in  $H$  is a centerpoint for  $Q$  in  $P$ . To see this take a chord  $s$  through  $c$  and let  $H_1, H_2, \dots$  be the half-polygons into which  $s$  splits  $P$ . Suppose for contradiction that some  $H_i$  contains more than  $2n/3$  points of  $Q$ . Let  $s_i \subseteq s$  be the part of  $s$  bounding  $H_i$ . Then, if we slightly shrink  $H_i$  by moving  $s_i$  infinitesimally, we obtain a half-polygon  $H'_i$  that contains more than  $2n/3$  vertices, contradicting the definition of  $c$ .

Our main result is the following:

**Theorem 15** Let  $Q$  be a set of  $n$  points inside a polygon (possibly with holes) with  $r$  reflex vertices. Then  $\mathcal{G}_{\text{vis}, P}^\times(Q)$  admits a clique-based separator of size  $O(\min(n, r) + \sqrt{n})$  and weight  $O(\min(n, r \log(n/r) + \sqrt{n}))$ . The bounds on the size and weight of the separator are tight in the worst case, even for simple polygons.

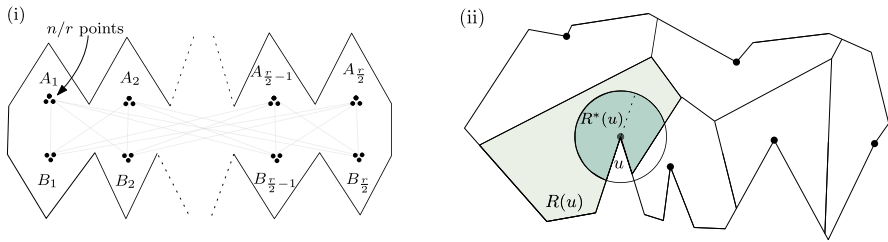
#### 4.1.1 The Lower Bound

Recall that even for non-visibility restricted unit-disk graphs,  $\Omega(\sqrt{n})$  is a lower bound on the worst-case size of the separator. Hence, to prove the lower bound of Theorem 15 it suffices to give an example where the size and weight are  $\Omega(\min(n, r))$  and  $\Omega(\min(n, r \log(n/r)))$ , respectively. This example is given in Fig. 5(i).

#### 4.1.2 The Upper Bound

We first describe our construction for polygons without holes. Our separator construction has two steps. In the first step we put all points that can see a reflex vertex within distance  $\sqrt{2}$  into the separator. We will argue that we can do this in such a way that we put  $O(1)$  cliques per reflex vertex into the separator. In the second step we handle the remaining points. We take a centerpoint  $c$  inside  $P$  and then define  $\sqrt{n}$  chords





**Fig. 5** (i) Each cluster  $A_i$  sees any of the clusters  $B_j$  completely and all distances are at most 1, so a separator that splits  $\mathcal{G}_{\text{vis}, P}^\times (\cup A_i \cup \cup B_j)$  into two or more components must fully contain  $\cup A_i$  or  $\cup B_j$ . Since the clusters  $A_i$  (and similarly  $B_j$ ) do not see each other, such a separator has size at least  $r/2$  and weight at least  $(r/2) \log(n/r)$ . (ii) Splitting  $R^*(u)$  into two convex parts

through  $c$ . For each chord  $s_i$  we put the points within distance  $1/2$  into the separator, suitably grouped into cliques. We will argue that the total weight of the cliques, over all chords  $s_i$  is  $O(n)$ . Hence, there is a chord whose cliques have total weight  $O(\sqrt{n})$ . Adding the weight of the cliques we added in Step 1 then gives us the desired separator.

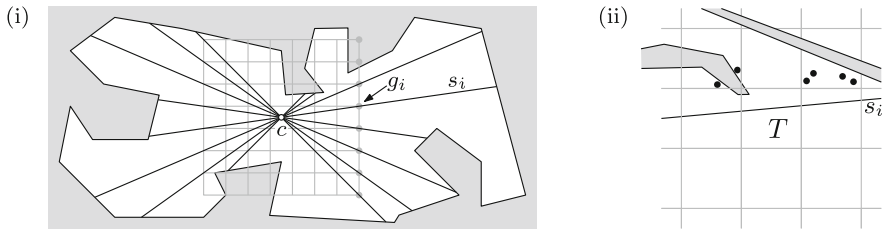
*Step 1: Handling points that see a nearby reflex vertex* Let  $V_{\text{ref}}$  be the set of reflex vertices of  $P$ , and let  $Q_1 \subseteq Q$  be the set of points that can see a reflex vertex within distance  $\sqrt{2}$ . Consider the geodesic Voronoi diagram of  $V_{\text{ref}}$  within  $P$ . Let  $R(u)$  be the Voronoi region of vertex  $u \in V_{\text{ref}}$  and define  $R^*(u) := R(u) \cap D(u, \sqrt{2})$ . Note that all points in  $R(u)$  can see  $u$  and that all points in  $Q_1$  are in  $R^*(u)$  for some vertex  $u \in V_{\text{ref}}$ . The following lemma is illustrated in Fig. 5(ii).

**Lemma 16** *Let  $e$  be an edge incident to  $u$ . If we extend  $e$  until it hits  $\partial R^*(u)$  then  $R^*(u)$  is split into two convex parts.*

**Proof** It suffices to argue that the angle of  $R(u)$  corresponding to  $u$  is the only reflex angle within  $R(u)$ . Then the result will follow from the fact that  $D(u, \sqrt{2})$  is convex. Clearly the shortest path connecting any point  $p \in R(u)$  to  $u$  is the segment  $pu$ . This implies that  $\partial R(u)$  consists of straight line segments that are parts of perpendicular bisectors of the segments connecting  $u$  to other reflex vertices of  $P$ . Any two such consecutive edges then cannot form a reflex angle within  $R(u)$ .

Since  $R^*(u)$  has diameter  $O(1)$ , Lemma 16 implies that  $Q_1 \cap R^*(u)$  can be partitioned into  $O(1)$  cliques. We collect all these cliques into a set  $\mathcal{S}_1$ . Since there are  $r$  reflex vertices,  $\mathcal{S}_1$  consists of  $O(r)$  cliques whose total weight is  $O(r \log(n/r))$ . Indeed, the total weight of these cliques is  $\sum_{C \in \mathcal{S}_1} \log(|C| + 1)$ , where  $|\mathcal{S}_1| = O(r)$  and  $\sum_{C \in \mathcal{S}_1} |C| \leq n$ . Jensen’s Inequality for concave functions implies that the sum of the weights is maximized when the clique sizes are all the same, which means that  $\sum_{C \in \mathcal{S}_1} \log(|C| + 1) = O(r \log(n/r))$ .

*Step 2: Handling points that do not see a nearby reflex vertex* Our separator  $\mathcal{S}$  consists of the cliques in  $\mathcal{S}_1$  plus a set  $\mathcal{S}_2$  of cliques that are found as follows. Let  $Q_2 := Q \setminus Q_1$  be the set of points that do not see a reflex vertex within distance  $\sqrt{2}$ . Let  $c$  be a centerpoint for  $Q_2$  inside  $P$ , that is, a point such that any chord through  $c$  splits  $P$  into two half-polygons containing at most  $2|Q_2|/3$  points from  $Q$ . Such a point exists by Theorem 14. Let  $G$  be a  $\sqrt{n} \times \sqrt{n}$  grid centered at  $c$ , where we assume



**Fig. 6** (i) The grid  $G$  defining the chords  $s_i$ . (ii) The points in the top-left cell do not see each other, but they see a reflex vertex and so they are not in  $Q_2(s_i)$ . The points in the top-right cell can be partitioned into  $O(1)$  cliques

for simplicity that  $\sqrt{n}$  is integer. For each of the  $\sqrt{n}$  points  $g_i$  in the rightmost column of the grid (even if  $g_i \notin P$ ), we define a maximal chord  $s_i$  by taking the line  $\ell_i$  through  $c$  and  $g_i$  and then taking the component of  $\ell_i \cap P$  that contains  $c$ ; see Fig. 6(i). We assume for simplicity that the chords  $s_i$  do not pass through reflex vertices; this can be ensured by slightly rotating the grid, if necessary.

For each chord  $s_i$ , we define  $Q_2(s_i)$  to be the set of points  $q \in Q_2$  such that there is a point  $z \in s_i$  that sees  $q$  with  $|qz| \leq 1/2$ .

Note that  $\mathcal{G}_{\text{vis}, P}^{\times}(Q)$  cannot have an arc between a point  $p \in Q_2 \setminus Q_2(s_i)$  above  $s_i$  and a point  $q \in Q_2 \setminus Q_2(s_i)$  below  $s_i$ ; otherwise  $p$  and/or  $q$  see a point on  $z$  within distance  $1/2$ , and so at least one of  $p, q$  is in  $Q_2(s_i)$ . Since  $s_i$  is a chord through the centerpoint  $c$ , this means that  $s_i$  induces a balanced separator.

It remains to argue that at least one of the chords  $s_i$  induces a separator whose weight is small enough. We will do this by creating a set  $\mathcal{S}(s_i)$  of cliques for each chord  $s_i$ , and prove that the total weight of these cliques, over all chords  $s_i$ , is  $O(n)$ . Since there are  $\sqrt{n}$  chords, one of them has the desired weight.

*Step 2.1: Points outside the grid* We simply put all points from  $Q_2(s_i)$  that lie outside the grid  $G$ —that is, the points that do not lie inside a grid cell—into  $\mathcal{S}(s_i)$ , as singletons.

**Observation 17** *Let  $q$  be a point that does not lie in any of the cells of the grid  $G$ . Then  $q$  lies within distance  $1/2$  of at most two chords  $s_i$ .*

Observation 17 implies that the total number of singleton cliques of points outside grid cells, summed over all chords  $\mathcal{S}(s_i)$ , is  $O(n)$ .

*Step 2.2: Points inside the grid* Next, consider a cell  $T$  of the grid  $G$ . Suppose a point  $q \in Q_2(s_i)$  sees a point  $z \in T \cap s_i$  such that  $|qz| \leq 1/2$ . Then  $q$  must lie inside one of the at most nine grid cells surrounding and including  $T$ . Consider such a cell  $T'$ .

**Lemma 18** *The points from  $Q_2(s_i) \cap T'$  that can see a point on  $s_i \cap T$  within distance  $1/2$  can be partitioned into  $O(1)$  cliques.*

**Proof** We will show that we can split these points into at most four cliques. To this end we split  $T'$  in four equal-sized subcells; see Fig. 7. Clearly each subcell has a diameter  $\frac{1}{2}\sqrt{2} < 1$ . It suffices now to show that if two points  $p, q$  in the same subcell

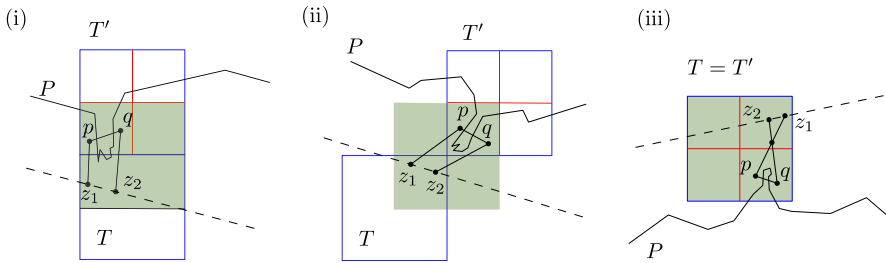


Fig. 7 Illustration for the proof of Lemma 18

do not see each other, then they see a reflex vertex within distance  $\sqrt{2}$ ; hence they are in  $Q_1$  and not in  $Q_2(s_i) \cap T'$ .

Let  $z_1$  and  $z_2$  be points on  $s_i \cap T$  that see  $p, q$ , respectively, with  $|pz_1| \leq 1/2$  and  $|qz_2| \leq 1/2$ . The quadrilateral formed by the points  $p, q, z_1, z_2$  must always lie within a square of side length 1. This square is shown in green in Fig. 7, for three different cases depending on the relative position of  $T'$  and  $T$ . In this quadrilateral, the side  $pq$  has to be crossed by  $\partial P$ , since we assumed  $p$  and  $q$  do not see each other. Moreover,  $\partial P$  cannot cross  $s_i$  or the segments  $pz_1, qz_2$ . Thus at least one reflex vertex of  $\partial P$  lies within the quadrilateral. It's then clear that both  $p, q$  will be able to see a reflex vertex within the quadrilateral (possibly not the same); since all points lie within a square of side length 1, the result follows. Note that in Fig. 7 (iii) we have made the edges  $pz_1, qz_2$  cross each other to showcase that the proof still goes through in this case.

We thus create  $O(1)$  cliques for each cell  $T'$  that is one of the at most nine cells surrounding a cell  $T$  crossed by  $s_i$ , and put them into  $\mathcal{S}(s_i)$ . This adds at most  $O(\log(n_{T'} + 1))$  weight to  $\mathcal{S}(s_i)$ , where  $n_{T'} := |Q_2 \cap T'|$ . Since there are  $n$  cells in total, this immediately gives a total weight of  $O(n \log n)$  over all sets  $\mathcal{S}(s_i)$ . Next we show that the the total weight of the cliques is actually  $O(n)$ . In what follows we denote by  $\text{weight}(\mathcal{S}(s_i))$  the total weight of the cliques corresponding to chord  $s_i$ .

**Lemma 19** *The total weight of the cliques corresponding to the chords  $s_i$  is  $O(n)$ , that is:*

$$\sum_{i=1}^{\sqrt{n}} \text{weight}(\mathcal{S}(s_i)) = O(n).$$

**Proof** Let  $\mathcal{T}(s_i)$  denote the cells crossed by a chord  $s_i$ , and let  $N[T]$  denote the at most nine cells surrounding a cell  $T$ , including  $T$  itself. By the arguments above, the total weight of all separators is.

$$\sum_{i=1}^{\sqrt{n}} \sum_{T \in \mathcal{T}(s_i)} \sum_{T' \in N[T]} O(\log(n_{T'} + 1)) \tag{1}$$

Let  $K_1$  be the set of cells in the central column of the grid, that is, the column containing the centerpoint  $c$ . For  $2 \leq j \leq m$ , where  $m := \lceil \sqrt{n}/2 \rceil$ , let  $K_j$  be the cells in the pair

of columns at distance  $j - 1$  from  $K_1$ . For example,  $K_2$  are the cells in the columns immediately to the left and right of  $K_1$ . Then we can make the following observations.

- The set  $K_j$  contains  $4j - 2 = O(j)$  cells that are crossed by at least one chord  $s_j$ .
- Cells in  $K_j$  are crossed by  $O(\sqrt{n}/j)$  chords.

Now define  $K_j^*$  to be the set of cells  $T'$  in  $K_j$  such that one of the nine cells surrounding  $T'$  is crossed by at least one chord. Note that each  $T' \in K_j^*$  is in  $N[T]$  for at most nine cells  $T$ , which are all in  $K_{j-1} \cup K_j \cup K_{j+1}$ . (Here we define  $K_{-1} = K_{m+1} = \emptyset$ .) It then follows from the above observations that  $|K_j^*| = O(j)$ . Moreover, if  $T' \in K_j^*$  and  $T' \in N[T]$ , then  $T$  is crossed by  $O(\sqrt{n}/j)$  chords. Hence, we can rewrite (1) as

$$\sum_{j=1}^m \left( O\left(\frac{\sqrt{n}}{j}\right) \cdot \sum_{T' \in K_j^*} O(\log(n_{T'} + 1)) \right) = O\left(\sum_{j=1}^m \frac{\sqrt{n}}{j} \cdot \sum_{T' \in K_j^*} \log(n_{T'} + 1)\right) \tag{2}$$

Now we are ready to compute the total weight of the separator. Note that

$$\sum_{T' \in K_j^*} \log(n_{T'} + 1) = \log\left(\prod_{T' \in K_j^*} (n_{T'} + 1)\right)$$

Now define  $n_j$  to be the total number of points in the cells in  $K_j^*$ . The AM-GM inequality gives us:

$$\log\left(\prod_{T' \in K_j^*} (n_{T'} + 1)\right) \leq \log\left(\left(\frac{\sum_{T' \in K_j^*} (n_{T'} + 1)}{|K_j^*|}\right)^{|K_j^*|}\right) = |K_j^*| \log\left(\frac{n_j}{|K_j^*|} + 1\right),$$

Since  $|K_j^*| = O(j)$  we have

$$|K_j^*| \log\left(\frac{n_j}{|K_j^*|} + 1\right) = O\left(j \cdot \log\left(\frac{n_j}{j} + 1\right)\right)$$

We conclude that the total weight over all separators is bounded by

$$O\left(\sum_{j=1}^m \frac{\sqrt{n}}{j} \cdot j \log\left(\frac{n_j}{j} + 1\right)\right) = O\left(\sum_{j=1}^m \sqrt{n} \cdot \log\left(\frac{n_j}{j} + 1\right)\right) \tag{3}$$

We have that:

$$\sum_{j=1}^m \log\left(\frac{n_j}{j} + 1\right) = \log\left(\frac{\prod_{j=1}^m (n_j + j)}{m!}\right) \tag{4}$$

Recall that  $m = \lceil \sqrt{n}/2 \rceil$ , which implies that  $n/m \leq 2m$ . Furthermore, Stirling’s approximation tells us that  $\ln(m!) = m \ln m - m + O(\ln m)$ . Changing the logarithm to base 2 we get  $\log(m!) = m \log m - m \log e + O(\log m)$ . Since  $\sum_{j=1}^m (n_j + j) \leq 3n$ , using the AM-GM inequality we derive

$$\begin{aligned} \log \left( \frac{\prod_{j=1}^m (n_j + j)}{m!} \right) &\leq \log \left( \left( \frac{3n}{m} \right)^m \right) - \log(m!) \\ &= m \log \left( \frac{3n}{m} \right) - (m \log m - m \log e + O(\log m)) \\ &\leq m \log(6m) - (m \log m - m \log e + O(\log m)) \\ &\leq 5m + O(\log m) \\ &= O(\sqrt{n}) \end{aligned}$$

Combing this with Equations (3) and (4) we conclude that the total weight of the sets  $\mathcal{S}(s)$  is  $O(n)$ , as claimed.

We conclude that there will be a chord  $s_i$  whose cliques have total weight  $O(\sqrt{n})$ . Then our desired separator is the set  $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}(s_i)$ .

### 4.2 Extension to Polygons With Holes

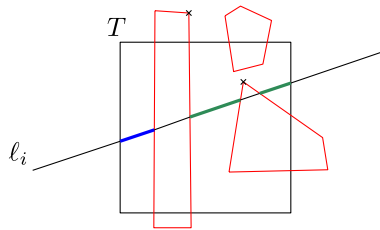
We now show how to extend the previous approach so that it also works for a polygon  $P$  with holes. The key ideas remain the same.

As previously, the first step is to put the points that see a reflex vertex within distance  $\sqrt{2}$  into the separator. This can again be done using the geodesic Voronoi diagram of the reflex vertices. Recall that a vertex of a hole is defined to be reflex if its angle within  $P$  (not: within the hole) is more than 180 degrees.

In the second step we again construct  $\sqrt{n}$  potential separators. When  $P$  has holes, a chord may not split  $P$  into two half-polygons. Hence, instead of taking chords though a centerpoint for  $Q_2$  in  $P$ , we proceed slightly differently. We take a regular centerpoint for  $Q_2$ , and for each point  $g_i$  in the rightmost column of the grid  $G$  we take the full line  $\ell_i$  through  $c$  and  $p_i$ . Note that  $\ell_i \cap P$  can consist of many chords. Together, these chords split  $P$  into several parts; the parts above  $\ell_i$  contain at most  $2|Q_2|/3$  points, and the parts below  $\ell_i$  contain at most  $2|Q_2|/3$  points as well. Hence, the resulting separators will be balanced.

Points in  $Q_2$  that lie outside the grid  $G$  can be handled as before. They are added as a singleton to  $O(1)$  lines  $\ell_i$ , so they contribute weight  $O(n)$  in total.

We now proceed to describe how to split points in  $Q_2$  within the grid into cliques. For each line  $\ell_i$  and each grid cell  $T$  we define an *entrance piece* to be the first piece of  $\ell_i \cap T \cap P$ . The remaining pieces will be referred to as *non-entrance pieces* (see Fig. 8). Entrance pieces can be handled exactly as in the case without holes: we look at all the points in the nine cells surrounding and including  $T$  which see this piece within a distance  $1/2$  and split them in a constant number of cliques. Then the total sum of the weights of these cliques for all the lines will be  $O(n)$  as before. Thus we can select a line  $\ell_i$  such that the weight of the cliques we add for the entrance pieces is  $O(\sqrt{n})$ .



**Fig. 8** A line  $\ell_i$  which crosses a cell  $T$  that has three holes, colored in red. The blue piece of  $\ell_i$  is the entrance piece, while the two green pieces are the non-entrance pieces. Note that each green piece can be uniquely associated to the pair of entrance-exit points of the hole preceding it. Marked with a cross are two possible choices of reflex vertices that are then associated with the green pieces

It remains now to handle the non-entrance pieces. Let  $\ell_i$  be the line we selected. We will show that  $\ell_i$  has  $O(r)$  non-entrance pieces and that for each of them we only need to add  $O(1)$  cliques. For the first part, observe that if a line  $\ell_i$  intersects a hole  $2k$  times then the hole must have at least  $k$  reflex vertices of  $P$ . These  $2k$  points of intersection can be split into  $k$  pairs of entrance-exit points. Now note that a non-entrance piece can be uniquely associated with a unique pair of entrance-exit points. Therefore we have associated each non-entrance piece with a unique reflex vertex of  $P$  and as a result the number of these pieces is indeed  $O(r)$ . The second part is handled as before by adding a constant number of cliques from the nine cells that surround the cell that contains each non-entrance piece. Therefore the cliques that correspond to these pieces add a total weight of  $O(r \log \frac{n}{r})$ .

To summarize, after having found a separator such that the total weight of the entrance pieces is  $O(\sqrt{n})$  we can add the cliques for the non-entrance pieces to obtain a separator of weight  $O(r \log \frac{n}{r} + \sqrt{n})$ . Together with the cliques from  $\mathcal{S}_1$  this gives the final separator. This finishes the proof of Theorem 15.

## 5 Applications

In this section, we show how one can use clique-based separators to get subexponential algorithms for several classic graph problems, namely MAXIMUM INDEPENDENT SET, FEEDBACK VERTEX SET, and  $q$ -COLORING for constant  $q$ . The algorithms are very generic: they only use the fact that the clique-based separators exist and can be computed in polynomial time, so they can immediately be combined with any of our separator theorems.

There are subexponential algorithms for MAXIMUM INDEPENDENT SET, FEEDBACK VERTEX SET, and 3-COLORING in the class of string graphs. In particular, Bonnet and Rzazewski [4] obtain a running time of  $2^{O(n^{2/3} \log n)}$  for MAXIMUM INDEPENDENT SET and 3-COLORING, and  $2^{O(n^{2/3} \log^{O(1)} n)}$  for FEEDBACK VERTEX SET. They also show that there is no subexponential algorithm for  $q$ -COLORING in string graphs in case of  $q \geq 4$ . All graph classes studied in this paper are a subclass of string graphs, therefore the algorithmic results of Bonnet and Rzazewski carry over. In the case of map graphs, one can derive  $2^{O(\sqrt{n} \log n)}$  algorithms

for MAXIMUM INDEPENDENT SET and FEEDBACK VERTEX SET by using the trivial bound  $k \leq n$  in the subexponential parameterized algorithms of Fomin et al. [14]. Moreover, the separator theorem of Matoušek [7] can be used to directly get a  $2^{O(\sqrt{n})}$  algorithm for  $q$ -COLORING for fixed constant  $q$ . Finally, for polygons of total complexity  $O(n)$ , one can use the parameterized algorithm of Marx and Philipczuk [22] to derive a  $2^{O(\sqrt{n} \log n)}$  algorithm for MAXIMUM INDEPENDENT SET.

Using our separator theorems, we get subexponential algorithms for the three problems mentioned above in the various graph classes under consideration. The running times we obtain match or slightly improve the results that can be obtained by applying the existing results mentioned above. It should be kept in mind, of course, that the existing results are for more general graph classes. An exception are our results on map graphs, which were explicitly studied before and where we improve the running time for MAXIMUM INDEPENDENT SET and FEEDBACK VERTEX SET from  $2^{O(\sqrt{n} \log n)}$  to  $2^{O(\sqrt{n})}$ . (Admittedly, the existing results apply in the parameterized setting while ours don't.) In any case, the main advantage of our approach is that it allows us to solve MAXIMUM INDEPENDENT SET, FEEDBACK VERTEX SET and  $q$ -COLORING on each of the mentioned graph classes in a uniform manner. Specifically, our result are as follows.

In map graphs that are given by a linear-sized representation we obtain  $2^{O(\sqrt{n})}$  algorithms for all three problems. In pseudodisk graphs and in intersection graphs of objects with linear union complexity, we get  $2^{O(n^{2/3} \log n)}$  algorithms for MAXIMUM INDEPENDENT SET and FEEDBACK VERTEX SET, and a  $2^{O(n^{2/3})}$  algorithm for  $q$ -COLORING with constant  $q$ . In intersection graphs of polygonal pseudodisks of total complexity  $O(n)$ , the obtained running times are  $2^{O(\sqrt{n} \log n)}$  for MAXIMUM INDEPENDENT SET and FEEDBACK VERTEX SET, and  $2^{O(\sqrt{n})}$  for  $q$ -COLORING with constant  $q$ . For intersection graphs of geodesic disks in a simple polygon, our separators yield the same running times as for pseudodisks:  $2^{O(n^{2/3} \log n)}$  algorithms for MAXIMUM INDEPENDENT SET and FEEDBACK VERTEX SET, and a  $2^{O(n^{2/3})}$  algorithm for  $q$ -COLORING with constant  $q$ .

We also get subexponential algorithms for each of our problems in visibility-restricted unit-disk graphs in polygons with  $r$  reflex vertices (assuming the total number of vertices is polynomial in  $n$ ). The running time is  $2^{f(n,r)}$ , where  $f(n,r) = O(\min(n, r \log(n/r)) + \sqrt{n})$ .

## 5.1 The Algorithms

Recall that the MAXIMUM INDEPENDENT SET problem is to find a maximum-size subset of pairwise non-adjacent nodes in a given graph  $G$ . The FEEDBACK VERTEX SET problem asks to find a minimum-size subset of nodes whose deletion makes  $G$  cycle-free. Here it will be more convenient to work with the complement of the solution, which corresponds to the MAXIMUM INDUCED FOREST problem: find a maximum-size subset of nodes that induce a forest in  $G$ . Finally, in the  $q$ -COLORING problem, the goal is to decide if  $G$  has a proper  $q$ -coloring (that is, a coloring of the nodes using  $q$  colors such that adjacent nodes have different colors). For  $q$ -COLORING, we



can immediately reject any instance that contains a clique of size at least  $q + 1$ , as such a clique has no proper  $q$ -coloring. Therefore the weight of each clique is at most  $\log(q + 1) = O(1)$ , and we only need to deal with instances of maximum clique size  $q$ . Consequently, if we have a clique-based separator  $\mathcal{S}$  with  $s$  cliques in the graph, then the weight of the separator is at most  $s \cdot \log(q + 1) = O(s)$ .

We can simplify the presentation by abstracting away the geometry. In what follows, we fix a graph class  $\mathcal{G}$ , and we assume that for any  $n$ -node graph  $G \in \mathcal{G}$  we can compute a balanced clique-based separator  $\mathcal{S}$  of  $G$  in polynomial time of size  $s_{\mathcal{G}}(n)$  and weight  $w_{\mathcal{G}}(n)$ . Note that our separator computations rely on having a geometric representation of the input graph  $G$ . We will prove the following theorem.

**Theorem 20** *Let  $\mathcal{G}$  be a class of geometric intersection graphs, and suppose that  $\mathcal{G}$  has a clique-based separator theorem of weight  $w_{\mathcal{G}}(n)$  and size  $s_{\mathcal{G}}(n) = n^c$  for some constant  $0 < c < 1$ . Suppose moreover that these separators can be computed in polynomial time. Then for an  $n$ -node graph  $G \in \mathcal{G}$  given by its geometric representation, one can solve:*

- (1) MAXIMUM INDEPENDENT SET in  $2^{O(w_{\mathcal{G}}(n))}$  time,
- (2) FEEDBACK VERTEX SET in  $2^{O(w_{\mathcal{G}}(n))}$  time,
- (3) and  $q$ -COLORING for fixed constant  $q$  in  $2^{O(s_{\mathcal{G}}(n))}$  time.

**Proof (1)** We use the same algorithm as De Berg et al. [11], which we recall here. Let  $X$  be a maximum independent set. Notice that any clique  $C$  can contain at most one node from  $X$ . Therefore, the number of ways that a given set  $X$  can intersect a clique-based separator  $\mathcal{S}$  is at most

$$\prod_{C \in \mathcal{S}} (|C| + 1) = 2^{\sum_{C \in \mathcal{S}} \log(|C| + 1)} = 2^{\text{weight}(\mathcal{S})}.$$

We can use a simple divide-and-conquer strategy to find the set  $X$ . We compute a separator  $\mathcal{S}$ , and for all independent sets  $X_{\mathcal{S}}$  in  $\bigcup_{C \in \mathcal{S}} C$ , we will find the largest independent set that  $X_{\mathcal{S}}$  is a part of. To this end we consider the graph  $G_{X_{\mathcal{S}}} := G[V \setminus (N(X_{\mathcal{S}}) \cup \bigcup_{C \in \mathcal{S}} C)]$ , that is, the graph  $G$  where we remove the nodes of the separator as well as those nodes that neighbor a node in  $X_{\mathcal{S}}$ . We compute a maximum independent set on each side of the separator, restricting ourselves to  $G_{X_{\mathcal{S}}}$ , in a recursive fashion. The correctness follows from the fact that the set  $X_{\mathcal{S}}^* := X \cap (\bigcup_{C \in \mathcal{S}} C)$  will be considered by the algorithm, and then the remainder of  $X$  on each side of the separator will form maximum independent sets within that side.

Since the number of sets  $X_{\mathcal{S}}$  to enumerate is at most  $2^{\text{weight}(\mathcal{S})} \leq 2^{w_{\mathcal{G}}(n)}$ , we get the following recursion for the running time:

$$T(n) = 2^{w_{\mathcal{G}}(n)} (2T(\beta n) + \text{poly}(n)),$$

where  $\beta < 1$  is the balance factor of the separator theorem. Solving this recursion yields that  $T(n) = 2^{O(w_{\mathcal{G}}(n))}$ , as required.

**(2)** We solve the complement problem, MAXIMUM INDUCED FOREST. Observe that an induced forest can have at most two nodes in each clique, therefore an induced

forest with node set  $X$  can intersect a clique-based separator  $S$  in at most

$$\prod_{C \in S} \left( \binom{|C|}{2} + |C| + 1 \right) < \prod_{C \in S} (|C| + 1)^2 = 2^{\sum_{C \in S} 2 \log(|C|+1)} = 2^{2 \cdot \text{weight}(S)}$$

many ways. However, unlike with MAXIMUM INDEPENDENT SET, the solutions on the two sides of the separator are not completely independent: we must ensure that their union is a forest, and does not induce any cycles. Let  $H$  be a set of nodes in  $G$ , called *boundary nodes*, and let  $\pi$  be a partition of  $H$ . We say that a forest  $X$  *realizes*  $\pi$  if  $H \subseteq X$  and  $x, y \in H$  are in the same connected component of  $G[X]$  if and only if they are in the same partition class of  $\pi$ .

A separator can contain up to  $O(|S|)$  nodes of the solution forest, which will become boundary nodes in  $A \cup S$  and  $B \cup S$  with some specific partitions. Formally, we need to solve the following problem: given a graph  $G$ , a boundary node set  $H$  and a partition  $\pi$ , find the maximum induced forest of  $G$  that realizes  $\pi$  on  $H$ . In order to keep the running time small, we need to ensure that this boundary set stays small compared to the size of the instance in the recursion tree. We observe that all of our separators can be extended to achieve balance with respect to any node set  $W \subset V(G)$ . Using such separators, we can ensure that the number of boundary nodes for an instance of size  $n$  during the recursion is at most  $O(s_G(n))$  by for example taking separators that are balanced with respect to all nodes and the boundary set  $H$  alternatingly. See [10] for a different way to keep the boundary set small. As  $H$  has  $|H|^{O(|H|)} = 2^{O(|H| \log |H|)} = 2^{O(s_G(n) \log n)}$  partitions, the running time recursion is

$$T(n) = 2^{2w_G(n) + O(s_G(n) \log n)} (T(\beta' n) + \text{poly}(n)),$$

where  $\beta' < 1$  is a constant that depends only on the separator balance factor  $\beta$  and the size function  $s_G$ . This gives a running time of

$$T(n) = 2^{O(w_G(n) + s_G(n) \log n)}.$$

The running time can be further reduced by using the rank-based approach of Bodlaender et al. [3]. Here instead of trying all partitions, one only needs to keep track of a so-called *representative set* of  $2^{O(|B|)}$  weighted partitions, and update these representative sets bottom-up in the recursion tree. We can implement this technique similarly to earlier implementations. See [11] for an example with maximum induced forest and [10] for using the technique in a separator-based divide-and-conquer algorithm) without a tree-decomposition. The resulting algorithm has the following running time recursion:

$$T(n) = 2^{2w_G(n) + O(s_G(n))} (T(\beta' n) + \text{poly}(n)) = 2^{O(w_G(n))} (T(\beta' n) + \text{poly}(n)).$$

Solving the recursion gives the desired running time of  $T(n) = 2^{O(w_G(n))}$ .

**(3)** We start by checking if there is a clique of size at least  $q + 1$ . This can be done by brute force in  $O(n^{q+1}) = \text{poly}(n)$  time. If such a clique is found then we can

reject. Otherwise, we use a divide-and-conquer strategy: for all proper  $q$ -colorings of the separator node set  $S = \bigcup_{C \in \mathcal{S}} C$ , we solve the problem recursively for the node sets  $S_1 = S \cup A$  and  $S_2 = S \cup B$ , where  $A$  and  $B$  are the two sides of the separator. Note that in order to make this work, we in fact solve a slightly extended problem, where some arbitrary subset of nodes are already colored, and the task is to extend this coloring to a proper  $q$ -coloring.

As cliques have size at most  $q = O(1)$ , a separator consisting of  $s_{\mathcal{G}}(n)$  cliques has  $O(s_{\mathcal{G}}(n))$  nodes, and its nodes can be colored in at most  $q^{O(s_{\mathcal{G}}(n))} = 2^{O(s_{\mathcal{G}}(n))}$  many ways. The running time obeys the recurrence

$$T(n) = 2^{O(s_{\mathcal{G}}(n))}(T(\beta'n) + \text{poly}(n)),$$

where  $\beta' < 1$  is a constant such that  $\max(|A|, |B|) + |S| < \beta'n$  is guaranteed by the separator for  $n$  large enough. The recursion solves to  $T(n) = 2^{O(s_{\mathcal{G}}(n))}$ .

## 6 Concluding Remarks

We showed how clique-based separators with sub-linear weight can be constructed for various classes of intersection graphs which involve non-fat objects. The main advantage of our approach is that we can solve different problems in the graph classes we study in a uniform manner. There are several natural questions that are left open. Some are listed below.

- *Improving the bound for geodesic disks and adding holes.* Our bound on geodesic disks is directly derived by our result on pseudo-disks. However, geodesic disks are much less general than pseudo-disks (and “closer” to regular disks). Hence, one would expect that the optimal weight is closer to  $O(\sqrt{n})$ . If we allow our polygon to have holes, then our approach for geodesic disks no longer works. Indeed, it is easy to see that even after applying our perturbation scheme, the resulting objects can intersect each other more than two times.
- *Improving the bound for pseudo-disks.* Regarding pseudo-disks, an interesting result [27] states that in every finite family of pseudo-disks in the plane one can find a “small” one, in the sense that it is intersected by only a constant number of disjoint pseudo-disks. This property is also shared by, for instance, convex fat objects. Does this mean that the two graph classes are related in some natural way? If yes, could this connection be exploited to construct separators with better bounds?

**Funding** The work in this paper is supported by the Dutch Research Council (NWO) through Gravitation-grant NETWORKS-024.002.003.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted

by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Aronov, B., de Berg, M.T., Ezra, E., Sharir, M.: Improved bounds for the union of locally fat objects in the plane. *SIAM J. Comput.* **43**(2), 543–572 (2014). <https://doi.org/10.1137/120891241>
2. Ben-Moshe, B., Hall-Holt, O., Katz, M.J., Mitchell, J.S.: Computing the visibility graph of points within a polygon. In: Proc. 20th ACM Symposium on Computational Geometry, pp. 27–35. ACM (2004). <https://doi.org/10.1145/997817.997825>
3. Bodlaender, H.L., Cygan, M., Kratsch, S., Nederlof, J.: Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth. *Inf. Comput.* **243**, 86–111 (2015). <https://doi.org/10.1016/j.ic.2014.12.008>
4. Bonnet, É., Rzazewski, P.: Optimality program in segment and string graphs. *Algorithmica* **81**(7), 3047–3073 (2019). <https://doi.org/10.1007/s00453-019-00568-7>
5. Breen, M.: A Helly-type theorem for intersections of compact connected sets in the plane. *Geom. Dedic.* **71**, 111–117 (1998). <https://doi.org/10.1023/A:1005003212822>
6. Chan, T.M.: Polynomial-time approximation schemes for packing and piercing fat objects. *J. Algorithms* **46**(2), 178–189 (2003). [https://doi.org/10.1016/S0196-6774\(02\)00294-8](https://doi.org/10.1016/S0196-6774(02)00294-8)
7. Chen, Z.Z.: Approximation algorithms for independent sets in map graphs. *J. Algorithms* **41**(1), 20–40 (2001). <https://doi.org/10.1006/jagm.2001.1178>
8. Chen, Z.Z., Grigni, M., Papadimitriou, C.H.: Map graphs. *J. ACM* **49**(2), 127–138 (2006). <https://doi.org/10.1145/506147.506148>
9. Clarkson, K.L., Shor, P.W.: Application of random sampling in computational geometry, II. *Discret. Comput. Geom.* **4**, 387–421 (1989). <https://doi.org/10.1007/BF02187740>
10. de Berg, M., Bodlaender, H.L., Kisfaludi-Bak, S., Kolay, S.: An ETH-tight exact algorithm for euclidean TSP. In: 59th IEEE Annual Symposium on Foundations of Computer Science (FOCS), pp. 450–461 (2018). <https://doi.org/10.1109/FOCS.2018.00050>
11. de Berg, M., Bodlaender, H.L., Kisfaludi-Bak, S., Marx, D., van der Zanden, T.C.: A framework for Exponential-Time-Hypothesis-tight algorithms and lower bounds in geometric intersection graphs. *SIAM J. Comput.* **49**, 1291–1331 (2020). <https://doi.org/10.1137/20M1320870>
12. de Berg, M., Cheong, O., van Kreveld, M.J., Overmars, M. H.: Computational geometry: algorithms and applications (3rd Ed). Springer, (2008). URL: <https://www.worldcat.org/oclc/227584184>
13. Djidjev, H., Venkatesan, S.M.: Reduced constants for simple cycle graph separation. *Acta Inform.* **34**(3), 231–243 (1997). <https://doi.org/10.1007/s002360050082>
14. Fomin, F. V., Lokshtanov, D., Panolan, F., Saurabh, S., Zehavi, M.: Decomposition of map graphs with applications. In: Proc. 46th International Colloquium on Automata, Languages, and Programming, (ICALP), pp. 60:1–60:15, (2019). <https://doi.org/10.4230/LIPIcs.ICALP.2019.60>
15. Fox, J., Pach, J., Tóth, C.D.: A bipartite strengthening of the crossing lemma. *J. Comb. Theory Ser. B* **100**(1), 23–35 (2010). <https://doi.org/10.1016/j.jctb.2009.03.005>
16. Har-Peled, S., Quanrud, K.: Approximation algorithms for polynomial-expansion and low-density graphs. *SIAM J. Comput.* **46**(6), 1712–1744 (2017). <https://doi.org/10.1137/16M1079336>
17. Kedem, K., Livne, R., Pach, J., Sharir, M.: On the union of Jordan regions and collision-free translational motion amidst polygonal obstacles. *Discret. Comput. Geom.* **1**, 59–70 (1986). <https://doi.org/10.1007/BF02187683>
18. Kisfaludi-Bak, S.: Hyperbolic intersection graphs and (quasi)-polynomial time. In: Proc. 31st ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1621–1638 (2020). <https://doi.org/10.1137/1.9781611975994.100>
19. Kisfaludi-Bak, S., Marx, D., van der Zanden, T. C.: How does object fatness impact the complexity of packing in  $d$  dimensions? In: 30th International Symposium on Algorithms and Computation, ISAAC 2019, vol. 149 of LIPIcs, pp. 36:1–36:18 (2019). <https://doi.org/10.4230/LIPIcs.ISAAC.2019.36>
20. Lee, J. R.: Separators in region intersection graphs. In: 8th Innovations in Theoretical Computer Science Conference, ITCS 2017, vol. 67 of LIPIcs, pp. 1:1–1:8 (2017). <https://doi.org/10.4230/LIPIcs.ITCS.2017.1>

21. Lipton, R.J., Tarjan, R.E.: A separator theorem for planar graphs. *SIAM J. Appl. Math.* **36**(2), 177–189 (1977). <https://doi.org/10.1137/0136016>
22. Marx, D., Pilipczuk, M.: Optimal parameterized algorithms for planar facility location problems using voronoi diagrams. In *Proc. 23rd Annual European Symposium on Algorithms (ESA)*, vol. 9294 of *Lecture Notes in Computer Science*, pp. 865–877. Springer (2015). [https://doi.org/10.1007/978-3-662-48350-3\\_72](https://doi.org/10.1007/978-3-662-48350-3_72)
23. Matousek, J.: *Lectures on Discrete Geometry*, vol. 212 of *Graduate Texts in Mathematics*, pp. 14–16. Springer (2002). <https://doi.org/10.1007/978-1-4613-0039-7>
24. Matoušek, J.: Near-optimal separators in string graphs. *Comb. Probab. Comput.* **23**(1), 135–139 (2014). <https://doi.org/10.1017/S0963548313000400>
25. Miller, G.L., Teng, S.H., Thurston, W.P., Vavasis, S.A.: Separators for sphere-packings and nearest neighbor graphs. *J. ACM* **44**(1), 1–29 (1997). <https://doi.org/10.1145/256292.256294>
26. Pach, J., Sharir, M.: Geometric incidences. In: Pach, J. (ed) *Towards a Theory of Geometric Graphs*, *Contemporary Mathematics*, vol. 342, pp. 185–223. American Mathematical Society (2004). <https://doi.org/10.1090/conm/342>
27. Pinchasi, R.: A finite family of pseudodiscs must include a “small” pseudodisc. *SIAM J. Discret. Math.* **28**, 1930–1934 (2014). <https://doi.org/10.1137/130949750>
28. Pollack, Ricky, Sharir, Micha, Rote, Günter.: Computing the geodesic center of a simple polygon. *Discret. Comput. Geom.* **4**(6), 611–626 (1989). <https://doi.org/10.1007/BF02187751>
29. Raman, R., Ray, S.: Constructing planar support for non-piercing regions. *Discret. Comput. Geom.* **64**, 1098–1122 (2020)
30. Smith, W. D., Wormald, N. C.: Geometric separator theorems & applications. In: *39th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 232–243. IEEE Computer Society (1998). <https://doi.org/10.1109/SFCS.1998.743449>

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