

Bipartizing (Pseudo-)Disk Graphs: Approximation with a Ratio Better than 3

Daniel Lokshtanov ✉ 

University of California, Santa Barbara, USA

Fahad Panolan ✉ 

University of Leeds, UK

Saket Saurabh ✉ 

Institute of Mathematical Sciences, India

Jie Xue ✉ 

New York University Shanghai, China

Meirav Zehavi ✉ 

Ben-Gurion University, Israel

Abstract

In a disk graph, every vertex corresponds to a disk in \mathbb{R}^2 and two vertices are connected by an edge whenever the two corresponding disks intersect. Disk graphs form an important class of geometric intersection graphs, which generalizes both planar graphs and unit-disk graphs. We study a fundamental optimization problem in algorithmic graph theory, BIPARTIZATION (also known as ODD CYCLE TRANSVERSAL), on the class of disk graphs. The goal of BIPARTIZATION is to delete a minimum number of vertices from the input graph such that the resulting graph is bipartite. A folklore (polynomial-time) 3-approximation algorithm for BIPARTIZATION on disk graphs follows from the classical framework of Goemans and Williamson [Combinatorica'98] for cycle-hitting problems. For over two decades, this result has remained the best known approximation for the problem (in fact, even for BIPARTIZATION on unit-disk graphs). In this paper, we achieve the first improvement upon this result, by giving a $(3 - \alpha)$ -approximation algorithm for BIPARTIZATION on disk graphs, for some constant $\alpha > 0$. Our algorithm directly generalizes to the broader class of *pseudo-disk* graphs. Furthermore, our algorithm is *robust* in the sense that it does not require a geometric realization of the input graph to be given.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases bipartization, geometric intersection graphs, approximation algorithms

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2024.6

Category APPROX

Related Version *Full Version*: <https://arxiv.org/abs/2407.09356>

1 Introduction

Disk graphs refer to intersection graphs of disks in the plane \mathbb{R}^2 . Formally, in a disk graph, every vertex corresponds to a disk in \mathbb{R}^2 and two vertices are connected by an edge whenever the two corresponding disks intersect. As a rather general class of geometric intersection graphs, disk graphs simultaneously generalize two important graph classes, unit-disk graphs and planar graphs, both of which have been extensively studied over decades. Many central problems in algorithmic graph theory have been considered on disk graphs, including VERTEX COVER [8, 28, 41], INDEPENDENT SET [8], MAXIMUM CLIQUE [5], FEEDBACK VERTEX SET [26, 28], DOMINATING SET [13], etc.



© Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Jie Xue, and Meirav Zehavi; licensed under Creative Commons License CC-BY 4.0
Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2024).

Editors: Amit Kumar and Noga Ron-Zewi; Article No. 6; pp. 6:1–6:14



Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In this paper, we investigate a fundamental optimization problem on the class of disk graphs, called BIPARTIZATION. In this problem, the input is a graph G and the goal is to delete a smallest number of vertices from G such that the resulting graph is bipartite. There has been a long line of work studying BIPARTIZATION (e.g., see [10, 20, 34, 36, 37] and citations therein). Observe that the edge counterpart of the BIPARTIZATION problem, EDGE BIPARTIZATION, where we need to find fewest edges whose deletion results in a bipartite graph, is equivalent to the classical MAXIMUM CUT problem (which has been studied for over five decades [14, 39]). It is known that EDGE BIPARTIZATION (and MAXIMUM CUT) reduces to BIPARTIZATION [42], and both problems find applications in computational biology [42, 38], VLSI chip design [18], genome sequence assembly [32], and more. BIPARTIZATION is of particular interest also due to the following observation: a graph is bipartite if and only if it does not contain any odd cycle. As such, it can be formulated as hitting all odd cycles in the graph (using fewest vertices). For this reason, the BIPARTIZATION problem is also known as ODD CYCLE TRANSVERSAL, and belongs to the family of *cycle-hitting* problems, one of the most well-studied topics in algorithmic graph theory. Besides BIPARTIZATION, other important cycle-hitting problems that have been studied on disk graphs include FEEDBACK VERTEX SET, TRIANGLE HITTING, SHORT CYCLE HITTING, etc.

There is no surprise that BIPARTIZATION is NP-complete. In fact, it is NP-complete even on graphs of maximum degree 3 and planar graphs of maximum degree 4 [6]. As such, the study of the problem is mainly in the context of approximation algorithms and parameterized algorithms. On the parameterized front, it was known that BIPARTIZATION can be solved in $2^{O(k)} \cdot n^{O(1)}$ time where k is the solution size [20, 25, 35]. On planar graphs and unit-disk graphs, there exist improved algorithms with running time $k^{O(\sqrt{k})} \cdot n^{O(1)}$ [2, 3, 29, 31], which are almost tight assuming the Exponential-Time Hypothesis (ETH). On disk graphs, it was not known whether BIPARTIZATION admits a subexponential-time parameterized algorithm, until very recently Lokshantov et al. gave a $k^{O(k^{27/28})} \cdot n^{O(1)}$ -time algorithm [26].

From the perspective of approximation algorithms (which is the focus of this paper), BIPARTIZATION is one of the trickiest problems in the sense that no polynomial-time approximation scheme (PTAS) was known on any (nontrivial) graph classes, but at the same time no inapproximability results was known except for general graphs. On general graphs, BIPARTIZATION cannot admit any (polynomial-time) constant-approximation algorithm assuming the Unique Games Conjecture [4], and the best known approximation ratio is $O(\sqrt{\log \text{opt}})$ due to Kratsch and Wahlström [24], improving the earlier bounds of $O(\log n)$ [9] and $O(\sqrt{\log n})$ [1]. It has been a long-standing open question whether BIPARTIZATION admits a PTAS on planar graphs or unit-disk graphs. On planar graphs, the currently best approximation for BIPARTIZATION is still the $\frac{9}{4}$ -approximation algorithm given in the seminal work of Goemans and Williamson [15] more than two decades ago. This result immediately gives a 3-approximation algorithm for BIPARTIZATION on disk graphs (and in particular, unit-disk graphs) by the well-known fact that triangle-free disk graphs are planar [22].

► **Theorem 1** (folklore). *There exists a polynomial-time 3-approximation algorithm for BIPARTIZATION on the class of disk graphs.*

Proof. Let G be the input disk graph. We repeat the following step until G contains no triangles: find a triangle in G , add its three vertices to the solution, and remove them from G . Denote by S the set of vertices added to the solution and by $G' = G - S$ the resulting triangle-free graph, which is planar [22]. Now apply the algorithm of Goemans and Williamson [15] on G' to obtain a $\frac{9}{4}$ -approximation solution S' . Note that $S \cup S'$ is a 3-approximation solution (for G). Indeed, any solution of BIPARTIZATION must hit all triangles in G and thus contains at least $|S|/3$ vertices in S . Also, it contains at least $|S'|/4$ vertices in $V(G')$. So its size is at least $|S|/3 + |S'|/4 \geq |S \cup S'|/3$. ◀

For over two decades, this has remained the best approximation algorithm for BIPARTIZATION on disk graphs (in fact, even on unit-disk graphs). Thus, there is a natural question to be asked: can we achieve an approximation ratio better than 3 for the problem? Note that one cannot achieve this by improving the approximation ratio $\frac{9}{4}$ for BIPARTIZATION on planar graphs. Indeed, even if we had a PTAS on planar graphs, the above argument still only gives us a 3-approximation algorithm on disk graphs. Therefore, the number 3 here is truly a bottleneck of the approximation ratio of the problem. In this paper, we break this bottleneck and answer the above question affirmatively by giving the first algorithm for BIPARTIZATION on disk graphs with an approximation ratio better than 3. Specifically, our main result is the following.

► **Theorem 2.** *There exists a polynomial-time $(3 - \alpha)$ -approximation algorithm for BIPARTIZATION on the class of disk graphs, for some constant $\alpha > 0$.*

We remark that Theorem 2 should be viewed as a proof of concept, rather than the quantitative improvement. Our algorithm in Theorem 2 is *robust* in the sense that it does not require the geometric realization of the input disk graph to be given. Furthermore, our algorithm directly generalizes to the broader class of *pseudo-disk* graphs, which are the intersection graphs of topological disks in which the boundaries of every two of them intersect at most twice. (Note that the 3-approximation algorithm in Theorem 1 also works for pseudo-disk graphs as triangle-free pseudo-disk graphs are planar [22].) Again, this generalized algorithm is robust.

► **Theorem 3.** *There exists a polynomial-time $(3 - \alpha)$ -approximation algorithm for BIPARTIZATION on the class of pseudo-disk graphs, for some constant $\alpha > 0$.*

Finally, we observe that our technique not only works for the BIPARTIZATION problem. In fact, it can be applied to a large category of vertex-deletion problems on (pseudo-)disk graphs, resulting in $(3 - \alpha)$ -approximation algorithms. Although at this point it only yields interesting results for BIPARTIZATION (mainly because most well-studied problems in the category already have algorithms on disk graphs with approximation ratio better than 3), this demonstrates that our technique could possibly have further applications in the future. We shall briefly discuss this part in Section 4.

Other related work

NP-complete optimization problems on disk graphs and other geometric intersection graphs have received considerable attention over years. Here we only summarize some recent work on this topic. The work of de Berg et al. [7] gave a framework for designing ETH-tight exact algorithms on (unit-)disk graphs or more generally (unit-)ball graphs, which works for a variety of classical optimization problems. Bonamy et al. [5] presented the first EPTAS and subexponential-time algorithm for MAXIMUM CLIQUE on disk graphs. Fomin et al. [11] designed almost ETH-tight parameterized algorithms for various cycle-packing and cycle-hitting problems on unit-disk graphs; in a follow-up paper [12], the same authors improved some of their algorithms to be ETH-tight. Recently, Lokshtanov et al. [26, 28] proposed frameworks for subexponential parameterized algorithms and EPTASes for various vertex-deletion problems on disk graphs (the framework for EPTASes does not work for BIPARTIZATION, while the one for subexponential parameterized algorithms works). A very recent work by the same authors [27] gave a 1.9999-approximation for VERTEX COVER on string graphs (i.e., intersection graphs of arbitrary connected geometric objects in the plane), which has the same flavor as this paper.

Besides the aforementioned algorithmic research on BIPARTIZATION, the problem was also studied in the context of kernelization complexity. The seminal work by Kratsch and Wahlström [23] showed that BIPARTIZATION admits a randomized polynomial kernel with respect to k . Later, for planar graphs, it was shown to admit a deterministic polynomial kernel by Jansen et al. [17]. Moreover, the kernelization complexity of BIPARTIZATION was studied also with respect to some structural parameterizations [16].

On a related note, we remark that structural properties of odd cycles in a graph has also received significant attention from various combinatorial points of view. While the survey of these results is beyond the scope of this paper, as an illustrative example, let us mention the study of Erdős–Pósa properties for odd cycles (see e.g., [10, 19, 33, 40]).

2 Preliminaries

Let G be a graph. We use $V(G)$ and $E(G)$ to denote the vertex set and edge set of G , respectively. For a subset $V \subseteq V(G)$, denote by $G[V]$ the subgraph of G induced by V , and by $G - V$ the subgraph of G induced by $V(G) \setminus V$. For a vertex v , we use $N_G(v)$ to denote the set $\{x \in V(G) \setminus \{v\} : (x, v) \in E(G)\}$. For a vertex subset S , we use $N_G(S)$ and $N_G[S]$ to denote the sets $\bigcup_{z \in S} N_G(z) \setminus S$ and $S \cup N_G(S)$, respectively. For a vertex v in G , we use $d_G(v)$ to denote the degree of v (i.e., $|N_G(v)|$) in G . A vertex subset $I \subseteq V(G)$ is a *distance- d independent* set in G , if for any two distinct vertices x and y in I , the distance between x and y in G is strictly more than d . Here, the distance between two vertices is the number of edges in a shortest path between those vertices. Let \mathcal{S} be a collection of subsets of $V(G)$. A *packing* of \mathcal{S} is a sub-collection \mathcal{S}' such that $S \cap S' = \emptyset$ for all $S, S' \in \mathcal{S}'$ with $S \neq S'$. We say a packing $\mathcal{S}' \subseteq \mathcal{S}$ is *maximal* if any $\mathcal{S}'' \subseteq \mathcal{S}$ satisfying $\mathcal{S}' \subsetneq \mathcal{S}''$ is not a packing of \mathcal{S} , and is *maximum* if any subset $\mathcal{S}'' \subseteq \mathcal{S}$ satisfying $|\mathcal{S}'| < |\mathcal{S}''|$ is not a packing of \mathcal{S} . Any maximum packing of \mathcal{S} has the same size.

An *odd cycle transversal* (or *OCT* for short) of G is a subset $S \subseteq V(G)$ such that $G - S$ is a bipartite graph. A *triangle* in G refers to a set $T = \{u, v, w\}$ of three vertices of G such that $(u, v), (v, w), (w, u) \in E(G)$. We use $\Delta(G)$ to denote the family of all triangles in G . For a set \mathcal{T} of triangles in G , we denote by $V(\mathcal{T})$ the set of vertices of all triangles in \mathcal{T} , i.e., $V(\mathcal{T}) = \bigcup_{T \in \mathcal{T}} T$. The notation $\text{tri}(\mathcal{T})$ denotes the size of a maximum packing of \mathcal{T} .

3 Our algorithm

In this section, we present our approximation algorithm for BIPARTIZATION on disk graphs. Our algorithm first takes a simple preprocessing step, which reduces the general problem to the problem on K_4 -free disk graphs (i.e., disk graphs without cliques of size 4). Then in the main part of our algorithm, we solve BIPARTIZATION on K_4 -free disk graphs. For a cleaner exposition, we shall present a randomized version of our algorithm, as it is more intuitive and yields a better approximation ratio. The derandomization can be found in the full version of the paper.

3.1 Preprocessing: reducing to the K_4 -free case

For a graph G , let $K_4(G)$ be the set of all K_4 's in G . The following lemma allows us to reduce the problem to BIPARTIZATION on K_4 -free disk graphs.

► **Lemma 4.** *Let G be a graph and \mathcal{C} be a packing of $K_4(G)$. Let S be a ρ -approximation solution for BIPARTIZATION on $G - V(\mathcal{C})$. Then, $S \cup V(\mathcal{C})$ is a $\max\{2, \rho\}$ -approximation solution for BIPARTIZATION on G .*

Proof. First, it is clear that $S \cup V(\mathcal{C})$ is an OCT of G , since $G - S \cup V(\mathcal{C}) = (G - V(\mathcal{C})) - S$ contains no odd cycles. Let S^* be an optimal OCT of G . For every K_4 in \mathcal{C} , S^* must contain at least 2 vertices in the K_4 (for otherwise $G - S^*$ contains a triangle). Since the K_4 's in \mathcal{C} are disjoint, we have $|S^* \cap V(\mathcal{C})| \geq 2|\mathcal{C}| = |V(\mathcal{C})|/2$. Furthermore, $S^* \cap (V(G) \setminus V(\mathcal{C}))$ is an OCT of $G - V(\mathcal{C})$, which implies $|S^* \cap (V(G) \setminus V(\mathcal{C}))| \geq |S|/\rho$. Therefore,

$$|S^*| = |S^* \cap V(\mathcal{C})| + |S^* \cap (V(G) \setminus V(\mathcal{C}))| \geq \frac{|V(\mathcal{C})|}{2} + \frac{|S|}{\rho} \geq \frac{|S| + |V(\mathcal{C})|}{\max\{2, \rho\}}.$$

As $|S \cup V(\mathcal{C})| = |S| + |V(\mathcal{C})|$, we have $|S \cup V(\mathcal{C})| \leq \max\{2, \rho\} \cdot |S^*|$. ◀

One reason for why this reduction helps us is the degeneracy of a K_4 -free disk graph. Recall that a graph G is *c-degenerate* if we can sort its vertices as v_1, \dots, v_n such that each v_i is neighboring to at most c vertices in $\{v_1, \dots, v_{i-1}\}$. A c -degenerate graph of n vertices has at most cn edges and thus has average degree at most $2c$. We prove that every K_4 -free disk graph is 11-degenerate. To prove this we use the following known result.

► **Lemma 5** ([30]). *Let D be a disk of radius r . Let \mathcal{S} be a set of pairwise non-overlapping disks of radius r such that every disk in \mathcal{S} intersects with D . Then, $|\mathcal{S}| \leq 5$.*

► **Lemma 6.** *Every K_4 -free disk graph is 11-degenerate.*

Proof. To prove the lemma, it is enough to prove that for any K_4 -free disk graph, there is a vertex of degree at most 11. Let G be a K_4 -free disk graph with a realization \mathcal{D} , and let $D_u \in \mathcal{D}$ be the disk representing the vertex $u \in V(G)$. Let u be a vertex in G such that disk D_u has the smallest radius among the disks in \mathcal{D} . We will prove that $d_G(u) \leq 11$. Let r be the radius of D_u . Notice that for each $v \in N_G(u)$, the radius of D_v is at least r . Now we construct a graph H with vertex set $N_G(u)$ such that H is a unit disk graph, H is a subgraph of G (and hence K_4 -free), and $d_G(u) = d_H(u)$. The construction of H is as follows. For each $v \in N_G(u)$, construct a disk D'_v of radius r which is fully contained in the disk D_v and intersects D_u . The graph H is the geometric intersection graph of $\mathcal{D}' = \{D_u\} \cup \{D'_v : v \in N_G(u)\}$. It is easy to see that H is a unit disk graph, H is a subgraph of G and $d_G(u) = d_H(u)$. For each $v \in V(H) \setminus \{u\}$, let L_v be the line segment between the centers of D_u and D'_v . Let $\{v_1, \dots, v_t\} = N_G(u)$ such that the line segments $L_{v_1}, L_{v_2}, \dots, L_{v_t}$ are in the clockwise order. We claim that $t \leq 11$. For the sake of contradiction assume that $t \geq 12$. Suppose there exists two distinct $i, j \in \{1, 3, 5, 7, 9, 11\}$ such that D'_{v_i} intersects with D'_{v_j} . This implies that $D'_{v_{i+1}}$ or $D'_{v_{j+1}}$ intersects both D'_{v_i} and D'_{v_j} . Let $w \in \{v_{i+1}, v_{j+1}\}$ be the vertex such that D'_w intersects both D'_{v_i} and D'_{v_j} . Then, $H[\{u, w, v_i, v_j\}]$ is a complete graph on 4 vertices, which is a contradiction because H is K_4 -free. Then, the disks $D'_{v_1}, D'_{v_3}, D'_{v_5}, D'_{v_7}, D'_{v_9}, D'_{v_{11}}$ are pairwise non-overlapping, which is a contradiction to Lemma 5. Thus, we proved that $t \leq 11$ and hence $d_H(v) = d_G(v) \leq 11$. That is, the degeneracy of G is at most 11. ◀

3.2 The main algorithm

Our main algorithm for BIPARTIZATION on K_4 -free disk graphs is presented in Algorithm 1. At the beginning, it takes an arbitrary maximal triangle packing \mathcal{T} of G (line 1) and defines \mathcal{O} as the triangles in G that have at least one vertex outside $V(\mathcal{T})$. In a high-level, our algorithm computes three different solutions S_1, S_2, S_3 and returns the best one.

The first solution S_1 is computed in exactly the same way as the 3-approximation algorithm described in the introduction. Specifically, we include in S_1 all vertices in the triangle packing \mathcal{T} , and an OCT X_1 of $G - V(\mathcal{T})$ computed by a sub-routine PLANARBIP (line 3), which is an algorithm for BIPARTIZATION on planar graphs.

6:6 Bipartizing (Pseudo-)Disk Graphs

The second solution S_2 is constructed in a more involved way. First, in each triangle $T \in \mathcal{T}$, we randomly sample one vertex $v_T \in T$ (line 6); here the function $\text{random}(T)$ returns each vertex in T with probability $\frac{1}{3}$ and different calls of random are independent. Let R be the vertices in $V(\mathcal{T})$ not sampled (line 7). Then we include in S_2 all vertices in R , all vertices in an arbitrary maximal triangle packing \mathcal{T}' of $G - R$ (line 8), and an OCT X_2 of $G - (R \cup V(\mathcal{T}'))$ computed by PLANARBIP.

If \mathcal{O} is nonempty, we need to construct our third solution S_3 . We take a maximal packing \mathcal{T}'' of the triangles in \mathcal{O} (line 12). Then we include in S_3 all vertices in $V(\mathcal{T}'') \cap V(\mathcal{T})$ and an OCT X_3 of $G - (V(\mathcal{T}'') \cap V(\mathcal{T}))$ recursively computed by our algorithm (line 13).

Finally, we return the best one among S_1, S_2, S_3 (line 15); here $\min\{S_1, S_2, S_3\}$ returns the set of smallest size among S_1, S_2, S_3 . If S_3 is not computed, we simply return $\min\{S_1, S_2\}$. It is obvious that each of S_1, S_2, S_3 is an OCT of G and thus the algorithm always returns a correct solution. The quality of the solution obtained will be analyzed in the next section. Also, one can easily see that Algorithm 1 runs in polynomial time. Indeed, except the recursive call of OCT in line 13, all the other steps can be done in polynomial time. Line 13 will only be executed when $\mathcal{O} \neq \emptyset$. In this case, $\mathcal{T}'' \neq \emptyset$ and $V(\mathcal{T}'') \neq \emptyset$. Thus, the graph $G - (V(\mathcal{T}'') \cap V(\mathcal{T}))$ has at most $n - 1$ vertices where $n = V(G)$, and the running time of Algorithm 1 satisfies the recurrence $T(n) \leq T(n - 1) + n^{O(1)}$ which solves to $T(n) = n^{O(1)}$.

■ **Algorithm 1** BIPARTIZATION(G). ▷ G is a K_4 -free disk graph

```

1:  $\mathcal{T} \leftarrow$  a maximal packing of  $\Delta(G)$ 
2:  $\mathcal{O} \leftarrow \{T \in \Delta(G) : T \not\subseteq V(\mathcal{T})\}$ 
3:  $X_1 \leftarrow \text{PLANARBIP}(G - V(\mathcal{T}))$  ▷ construct the first solution  $S_1$ 
4:  $S_1 \leftarrow V(\mathcal{T}) \cup X_1$ 
5: for every  $T \in \mathcal{T}$  do ▷ construct the second solution  $S_2$ 
6:    $v_T \leftarrow \text{random}(T)$ 
7:    $R \leftarrow \bigcup_{T \in \mathcal{T}} (T \setminus \{v_T\})$ 
8:    $\mathcal{T}' \leftarrow$  a maximal packing of  $\Delta(G - R)$ 
9:    $X_2 \leftarrow \text{PLANARBIP}(G - (R \cup V(\mathcal{T}')))$ 
10:  $S_2 \leftarrow R \cup V(\mathcal{T}') \cup X_2$ 
11: if  $\mathcal{O} \neq \emptyset$  then ▷ construct the third solution  $S_3$ 
12:    $\mathcal{T}'' \leftarrow$  a maximal packing of  $\mathcal{O}$ 
13:    $X_3 \leftarrow \text{BIPARTIZATION}(G - (V(\mathcal{T}'') \cap V(\mathcal{T})))$ 
14:    $S_3 \leftarrow (V(\mathcal{T}'') \cap V(\mathcal{T})) \cup X_3$ 
15: return  $\min\{S_1, S_2, S_3\}$  ▷ if  $S_3$  is undefined, simply return  $\min\{S_1, S_2\}$ 

```

3.3 Analysis

In this section, we analyze the (expected) approximation ratio of Algorithm 1. We denote by ρ this ratio and aim to establish an upper bound for ρ . Consider a given disk graph G which is K_4 -free. Let opt be the minimum size of an odd cycle transversal of G , and \mathcal{T}, \mathcal{O} be the two sets of triangles as in Algorithm 1. The output of Algorithm 1 is the best one among three OCT solutions S_1, S_2, S_3 . Therefore, to analyze the approximation ratio of our algorithm, we have to consider the approximation ratios of S_1, S_2, S_3 . It turns out that each solution S_i individually may be of size 3opt or even larger in worst case. However, as we will see, the best one among them always admits an approximation ratio strictly smaller than 3.

In order to analyze the three solutions, we define two important parameters: $a = |\mathcal{T}|/\text{opt}$ and $b = \text{tri}(\mathcal{O})/\text{opt}$ (recall that $\text{tri}(\mathcal{O})$ is the size of a *maximum* packing of \mathcal{O}). Note that $a, b \in [0, 1]$ because both $|\mathcal{T}|$ and $\text{tri}(\mathcal{O})$ are at most the size of a maximum triangle packing in G , which is smaller than or equal to opt . The analysis of S_1, S_2, S_3 will be done in terms of a and b , that is, we shall represent the approximation ratios of S_1, S_2, S_3 as functions of a and b . Roughly speaking, we shall show that S_1 is good when a is small, S_2 is good when b is small, and S_3 is good when both a and b are large. For convenience, we use the notation ρ_0 to denote the approximation ratio of the PLANARBIP sub-routine used in Algorithm 1.

3.3.1 The quality of S_1

The solution S_1 is computed using exactly the 3-approximation algorithm described in the introduction. A more careful analysis shows that its approximation ratio is related to the parameter a : the smaller a is, the better S_1 is.

► **Lemma 7.** $|S_1| \leq (3a + \rho_0(1 - a)) \cdot \text{opt}$.

Proof. Since $a = |\mathcal{T}|/\text{opt}$, and \mathcal{T} is a triangle packing, we get that $|V(\mathcal{T})| = 3|\mathcal{T}| = 3a \cdot \text{opt}$. Since \mathcal{T} is a triangle packing, any odd cycle transversal contains at least $|\mathcal{T}|$ vertices from $V(\mathcal{T})$, the size of a minimum odd cycle transversal in $G - V(\mathcal{T})$ is at most $\text{opt} - |\mathcal{T}| = (1 - a)\text{opt}$. Therefore, $|S_1| = |V(\mathcal{T})| + |X_1| \leq (3a + \rho_0(1 - a)) \cdot \text{opt}$. ◀

3.3.2 The quality of S_2

Figuring out the quality of S_2 is the most involved part in our analysis. Basically, what we shall show is that whenever the parameter b is sufficiently small, S_2 always gives us a better-than-3 approximation no matter what the value of a is. The analysis in this section shall explicitly use the fact that G is K_4 -free. Let v_T, R , and \mathcal{T}' be as in Algorithm 1. We first need the following simple observation, which will allow us to bound the expected size of S_2 using the expected size of \mathcal{T}' .

► **Observation 8.** $\mathbb{E}[|R \cap S_{\text{opt}}|] \geq \frac{1}{3}|R|$ and $|V(\mathcal{T}') \cap S_{\text{opt}}| \geq |\mathcal{T}'|$.

Proof. Since S_{opt} is an OCT of G , it contains at least one vertex in every triangle $T \in \mathcal{T}$. Thus, $\mathbb{E}[|(T \setminus \{v_T\}) \cap S_{\text{opt}}|] \geq \frac{2}{3}$. By the linearity of expectation, we have

$$\mathbb{E}[|R \cap S_{\text{opt}}|] = \sum_{T \in \mathcal{T}} \mathbb{E}[|(T \setminus \{v_T\}) \cap S_{\text{opt}}|] \geq \frac{2}{3}|\mathcal{T}| = \frac{1}{3}|R|.$$

The fact $|V(\mathcal{T}') \cap S_{\text{opt}}| \geq |\mathcal{T}'|$ follows from the fact that \mathcal{T}' is a triangle packing: S_{opt} contains at least one vertex in every triangle $T \in \mathcal{T}'$. ◀

The above observation implies that $\mathbb{E}[|(R \cup V(\mathcal{T}') \cap S_{\text{opt}})|] \geq \frac{1}{3}|R| + \mathbb{E}[|\mathcal{T}'|]$. Therefore, we have $\mathbb{E}[|S_{\text{opt}} \setminus (R \cup V(\mathcal{T}'))|] \leq \text{opt} - \frac{1}{3}|R| - \mathbb{E}[|\mathcal{T}'|]$ and hence $\mathbb{E}[|X_2|] \leq \rho_0 \cdot (\text{opt} - \frac{1}{3}|R| - \mathbb{E}[|\mathcal{T}'|])$. It follows that

$$\begin{aligned} \mathbb{E}[|S_2|] &= |R| + \mathbb{E}[|V(\mathcal{T}')|] + \mathbb{E}[|X_2|] \\ &\leq |R| + 3 \cdot \mathbb{E}[|\mathcal{T}'|] + \rho_0 \cdot \left(\text{opt} - \frac{1}{3}|R| - \mathbb{E}[|\mathcal{T}'|] \right) \\ &= \left(1 - \frac{\rho_0}{3} \right) \cdot |R| + \rho_0 \cdot \text{opt} + (3 - \rho_0) \cdot \mathbb{E}[|\mathcal{T}'|] \end{aligned} \tag{1}$$

6:8 Bipartizing (Pseudo-)Disk Graphs

We say a vertex $v \in V(\mathcal{T})$ is *dead* if $v \notin R$ and v is not contained in any triangle in $G[V(\mathcal{T}) \setminus R]$. Let D denote the set of all dead vertices, which is a random subset of $V(\mathcal{T})$ as it depends on the random vertices v_T . For each $v \in V(\mathcal{T})$, let $\deg(v)$ denote the degree of v in $G[V(\mathcal{T})]$. Recall that $a = |\mathcal{T}|/\text{opt}$ and $b = \text{tri}(\mathcal{O})/\text{opt}$. It is easy to see the following relation between $|\mathcal{T}'|$ and $|D|$.

► **Observation 9.** $|\mathcal{T}'| \leq (a + b) \cdot \text{opt} - \frac{|R|}{3} - \frac{|D|}{3}$.

Proof. Since \mathcal{T}' is a triangle packing, we have $|\mathcal{T}' \cap \mathcal{O}| \leq \text{tri}(\mathcal{O}) = b \cdot \text{opt}$. On the other hand, all elements in $\mathcal{T}' \setminus \mathcal{O}$ are triangles in $G[V(\mathcal{T}) \setminus R]$. However, dead vertices cannot be the vertex of any triangle in $G[V(\mathcal{T}) \setminus R]$. Thus, the vertices of the triangles in $\mathcal{T}' \setminus \mathcal{O}$ must lie in $V(\mathcal{T}) \setminus (R \cup D)$. We have $|V(\mathcal{T}) \setminus (R \cup D)| = 3 \cdot |\mathcal{T}| - |R| - |D| = 3a \cdot \text{opt} - |R| - |D|$, which implies $|\mathcal{T}' \setminus \mathcal{O}| \leq a \cdot \text{opt} - \frac{|R|}{3} - \frac{|D|}{3}$. Because $|\mathcal{T}'| = |\mathcal{T}' \cap \mathcal{O}| + |\mathcal{T}' \setminus \mathcal{O}|$, we have the inequality in the observation. ◀

Combining the above observation with Equation 1, we have

$$\begin{aligned} \mathbb{E}[|S_2|] &\leq \left(1 - \frac{\rho_0}{3}\right) \cdot |R| + \rho_0 \cdot \text{opt} + (3 - \rho_0) \cdot \mathbb{E}[|\mathcal{T}'|] \\ &= \left(1 - \frac{\rho_0}{3}\right) \cdot |R| + \rho_0 \cdot \text{opt} + (3 - \rho_0) \cdot \left((a + b) \cdot \text{opt} - \frac{|R|}{3} - \frac{\mathbb{E}[|D|]}{3}\right) \\ &= (\rho_0 + (3 - \rho_0)(a + b)) \cdot \text{opt} - (3 - \rho_0) \cdot \frac{\mathbb{E}[|D|]}{3}. \end{aligned} \quad (2)$$

To show that S_2 has an approximation ratio below 3 when b is small, the crucial observation is that we have a large number of dead vertices in expectation.

► **Observation 10.** $\Pr[v \text{ is dead}] \geq \left(\frac{1}{3}\right)^{\frac{3}{8}\deg(v) + \frac{1}{4}}$ for all $v \in V(\mathcal{T})$. Thus, we have $\mathbb{E}[|D|] \geq \left(\frac{1}{3}\right)^{\frac{3}{8}d + \frac{1}{4}}(3a \cdot \text{opt})$, where d is the average degree of $G[V(\mathcal{T})]$.

Proof. Let $v \in V(\mathcal{T})$ and $T_0 \in \mathcal{T}$ be the triangle containing v . Denote by $N(v)$ the set of neighbors of v in $V(\mathcal{T})$ (excluding v itself). We then have $|N(v)| = \deg(v)$ and $|N(v) \setminus T_0| = \deg(v) - 2$. Observe that the graph $G[N(v)]$ is triangle-free. Indeed, if $G[N(v)]$ contains a triangle T , then $T \cup \{v\}$ forms a clique in G of size 4, which contradicts with the fact that G is K_4 -free. As $G[N(v)]$ is triangle-free, it is planar. In particular, $G[N(v) \setminus T_0]$ is planar. It was known that every n -vertex planar graph has a vertex cover of size at most $\frac{3}{4}n$ (indeed, a planar graph is 4-colorable, so it has an independent set of size at least $\frac{n}{4}$ and thus a vertex cover of size at most $\frac{3}{4}n$). Thus, $G[N(v) \setminus T_0]$ has a vertex cover $C \subseteq N(v) \setminus T_0$ with $|C| \leq \frac{3}{4}|N(v) \setminus T_0| = \frac{3}{4}(\deg(v) - 2)$.

Next, we notice that if $v \notin R$ and $C \subseteq R$, then v is a dead vertex. Indeed, if v is contained in a triangle $\{v, u, w\}$ in $G[V(\mathcal{T}) \setminus R]$, then at least one of u and w must be in C , since C is a vertex cover of $N(v) \setminus T_0$ (note that $u, w \notin T_0$ for otherwise $v \in R$). Let $\mathcal{T}_1 \subseteq \mathcal{T}$ (resp., $\mathcal{T}_2 \subseteq \mathcal{T}$) consist of the triangles that contain one vertex (resp., two vertices) in C . Note that no triangle in \mathcal{T} can contain three vertices in C because $G[C]$ is triangle-free. Therefore, $C \subseteq R$ if and only if $v_T \notin C$ for all $T \in \mathcal{T}_1 \cup \mathcal{T}_2$. The events $v_T \notin C$ for all $T \in \mathcal{T}_1 \cup \mathcal{T}_2$ are independent, and happen with probability $\frac{2}{3}$ if $T \in \mathcal{T}_1$ and with probability $\frac{1}{3}$ if $T \in \mathcal{T}_2$. It follows that $\Pr[C \subseteq R] = \left(\frac{2}{3}\right)^{|\mathcal{T}_1|} \cdot \left(\frac{1}{3}\right)^{|\mathcal{T}_2|}$. We have the inequality $|\mathcal{T}_1| + 2|\mathcal{T}_2| = |C| \leq \frac{3}{4}(\deg(v) - 2)$. Subject to $|\mathcal{T}_1| \geq 0$, $|\mathcal{T}_2| \geq 0$, and $|\mathcal{T}_1| + 2|\mathcal{T}_2| \leq \frac{3}{4}(\deg(v) - 2)$, the quantity $\left(\frac{2}{3}\right)^{|\mathcal{T}_1|} \cdot \left(\frac{1}{3}\right)^{|\mathcal{T}_2|}$ is minimized when $|\mathcal{T}_1| = 0$ and $|\mathcal{T}_2| = \frac{3}{8}(\deg(v) - 2)$, and is equal to $\left(\frac{1}{3}\right)^{\frac{3}{8}(\deg(v) - 2)}$. Thus, we have $\Pr[C \subseteq R] \geq \left(\frac{1}{3}\right)^{\frac{3}{8}(\deg(v) - 2)}$. Furthermore, the events $v \notin R$ and $C \subseteq R$ are independent because whether $v \notin R$ happens only depends on the choice of $v_{T_0} \in T_0$. We have $\Pr[v \notin R] = \frac{1}{3}$ and $\Pr[C \subseteq R] \geq \left(\frac{1}{3}\right)^{\frac{3}{8}(\deg(v) - 2)}$. Since v is a dead vertex if both $v \notin R$ and $C \subseteq R$ happen, we finally have $\Pr[v \text{ is dead}] \geq \left(\frac{1}{3}\right)^{\frac{3}{8}\deg(v) + \frac{1}{4}}$. By the linearity of expectation and the fact $|V(\mathcal{T})| = 3|\mathcal{T}| = 3a \cdot \text{opt}$, we then have

$$\mathbb{E}[|D|] = \sum_{v \in V(\mathcal{T})} \Pr[v \text{ is dead}] \geq \sum_{v \in V(\mathcal{T})} \left(\frac{1}{3}\right)^{\frac{3}{8}\deg(v) + \frac{1}{4}} \geq \left(\frac{1}{3}\right)^{\frac{3}{8}d + \frac{1}{4}} (3a \cdot \text{opt}).$$

where $d = \sum_{v \in V(\mathcal{T})} \deg(v) / |V(\mathcal{T})|$. ◀

At the end, we can establish the bound for $\mathbb{E}[|S_2|]$.

► **Lemma 11.** $\mathbb{E}[|S_2|] \leq (\rho_0 + (1 - (\frac{1}{3})^{\frac{3}{8}d + \frac{1}{4}})(3 - \rho_0)a + (3 - \rho_0)b) \cdot \text{opt}$, where d denotes the average degree of $G[V(\mathcal{T})]$.

Proof. Combining Observation 10 with Equation 2 completes the proof. ◀

According to Lemma 6, we have $d \leq 22$, and thus the above lemma gives us a good bound for $\mathbb{E}[|S_2|]$: as long as b is sufficiently small, no matter what a is, we can have that $\mathbb{E}[|S_2|] \leq (3 - \alpha) \cdot \text{opt}$ for some constant $\alpha > 0$.

3.3.3 The quality of S_3

Finally, we analyze the quality of S_3 . Given S_1 is good when a is small and S_2 is good when b is small, we clearly want S_3 to be good when both a and b are large.

► **Lemma 12.** *If $\rho \geq 2$, then $\mathbb{E}[|S_3|] \leq (\frac{2b}{3} + \rho(2 - a - \frac{b}{3})) \cdot \text{opt}$.*

Proof. Let $r = |\mathcal{T}''| / \text{opt}$. Since \mathcal{T}'' is a *maximal* packing in \mathcal{O} , $V(\mathcal{T}'')$ is a hitting set of \mathcal{O} , which implies $3|\mathcal{T}''| = |V(\mathcal{T}'')| \geq \text{tri}(\mathcal{O})$ and thus $r \geq \frac{\text{tri}(\mathcal{O})}{3 \cdot \text{opt}} = \frac{b}{3}$.

▷ **Claim 13.** $\mathbb{E}[|X_3|] \leq \rho(2 - a - r)\text{opt}$.

Proof. Let S^* be an optimal OCT of G . So $|S^*| = \text{opt}$. We call a triangle $T \in \mathcal{T}''$ *bad* if S^* contains a vertex from $V(T) \setminus V(\mathcal{T})$. Since \mathcal{T} is a triangle packing, we also know that $|S^* \cap V(\mathcal{T})| \geq |\mathcal{T}| = a \cdot \text{opt}$. Thus, the number of bad triangles in \mathcal{T}'' is at most $(1 - a)\text{opt}$, because S^* contains at most $(1 - a)\text{opt}$ vertices outside of $V(\mathcal{T})$, and any such vertex can be part of at most one triangle in \mathcal{T}'' . That is, the number of *good* triangles in \mathcal{T}'' is at least $(r - (1 - a))\text{opt}$. For each good triangle $T \in \mathcal{T}''$, S^* contains a vertex from $V(T) \cap V(\mathcal{T})$. This implies that $|S^* \cap (V(\mathcal{T}'') \cap V(\mathcal{T}))| \geq (r - (1 - a))\text{opt}$, and hence $|S^* \setminus (V(\mathcal{T}'') \cap V(\mathcal{T}))| \leq \text{opt} - (r - (1 - a))\text{opt} = (2 - a - r)\text{opt}$. Also, notice that $S^* \setminus (V(\mathcal{T}'') \cap V(\mathcal{T}))$ is an OCT of $G - (V(\mathcal{T}'') \cap V(\mathcal{T}))$. Thus, the size of an optimal OCT of $G - (V(\mathcal{T}'') \cap V(\mathcal{T}))$ is at most $(2 - a - r)\text{opt}$, which implies $\mathbb{E}[|X_3|] \leq \rho(2 - a - r)\text{opt}$. ◀

Since each triangle $T \in \mathcal{T}''$ is also in \mathcal{O} , we have $|V(T) \cap V(\mathcal{T})| \leq 2$. This implies that $|V(\mathcal{T}'') \cap V(\mathcal{T})| \leq 2r \cdot \text{opt}$. Now we are ready to deduce

$$\begin{aligned} \mathbb{E}[|S_3|] &= \mathbb{E}[|V(\mathcal{T}'') \cap V(\mathcal{T})| + |X_3|] \\ &= |V(\mathcal{T}'') \cap V(\mathcal{T})| + \mathbb{E}[|X_3|] \\ &\leq (2r \cdot \text{opt}) + \rho(2 - a - r)\text{opt} \quad (\text{By Claim 13 and } |V(\mathcal{T}'') \cap V(\mathcal{T})| \leq 2r \cdot \text{opt}) \\ &= (2 - \rho)r \cdot \text{opt} + \rho(2 - a)\text{opt} \\ &\leq (2 - \rho)\frac{b}{3}\text{opt} + \rho(2 - a)\text{opt} \quad (\text{Because } r \geq \frac{b}{3}, \text{ and } 2 - \rho < 0) \\ &\leq \left(\frac{2b}{3} + \rho\left(2 - a - \frac{b}{3}\right)\right) \cdot \text{opt}. \end{aligned}$$

This completes the proof of the lemma. ◀

3.3.4 Putting everything together

Given the analyses for S_1 , S_2 , and S_3 in the previous sections, we are ready to bound the (expected) approximation ratio ρ of the entire algorithm. Let $\rho_1 = 3a + \rho_0(1 - a)$, $\rho_2 = \rho_0 + (1 - (\frac{1}{3})^{\frac{3}{8}d + \frac{1}{4}})(3 - \rho_0)a + (3 - \rho_0)b$, $\rho_3 = \frac{2b}{3} + \rho(2 - a - \frac{b}{3})$ be the approximation ratios of S_1, S_2, S_3 given in Lemmas 7, 11, 12, respectively¹. As the output is the best one among S_1, S_2, S_3 , we have

$$\rho \leq \frac{\mathbb{E}[\min\{|S_1|, |S_2|, |S_3|\}]}{\text{opt}} \leq \frac{\min\{\mathbb{E}[|S_1|], \mathbb{E}[|S_2|], \mathbb{E}[|S_3|]\}}{\text{opt}} \leq \min\{\rho_1, \rho_2, \rho_3\}.$$

Note that ρ_1, ρ_2, ρ_3 can be viewed as linear functions of a and b , when the other numbers d, ρ_0, ρ are all fixed. So we first figure out the values of a and b that maximizes $\min\{\rho_1, \rho_2, \rho_3\}$. With calculation, we have

$$\min\{\rho_1, \rho_2, \rho_3\} \leq \frac{(3 - \rho_0)(2\rho - \rho_0)}{(3 - \rho_0 + \rho) + (\rho - 2) \cdot 3^{-(\frac{3}{8}d + \frac{5}{4})}} + \rho_0, \quad (3)$$

and the upper bound is achieved when

$$a = \frac{(2\rho - \rho_0)}{(3 - \rho_0 + \rho) + (\rho - 2) \cdot 3^{-(\frac{3}{8}d + \frac{5}{4})}} \quad \text{and} \quad b = \frac{(2\rho - \rho_0) \cdot 3^{\frac{3}{8}d + \frac{1}{4}}}{(3 - \rho_0 + \rho) + (\rho - 2) \cdot 3^{-(\frac{3}{8}d + \frac{5}{4})}},$$

in which case $\rho_1 = \rho_2 = \rho_3$. Now combine Equation 3 with the inequality $\rho \leq \min\{\rho_1, \rho_2, \rho_3\}$ and re-arrange the terms in the inequality, we deduce

$$(3^{-(\frac{3}{8}d + \frac{5}{4})} + 1) \cdot \rho^2 - ((2 + \rho_0) \cdot 3^{-(\frac{3}{8}d + \frac{5}{4})} + 3) \cdot \rho + (2\rho_0 \cdot 3^{-(\frac{3}{8}d + \frac{5}{4})}) \leq 0.$$

The left-hand side of the above inequality is a quadratic function of ρ in which the coefficient of the quadratic term is positive. Therefore, in order to make the quadratic function non-positive, ρ must be smaller than its larger root, i.e.,

$$\rho \leq \frac{((2 + \rho_0) \cdot 3^{-(\frac{3}{8}d + \frac{5}{4})} + 3) + \sqrt{((2 + \rho_0) \cdot 3^{-(\frac{3}{8}d + \frac{5}{4})} + 3)^2 - (3^{-(\frac{3}{8}d + \frac{5}{4})} + 1)(8\rho_0 \cdot 3^{-(\frac{3}{8}d + \frac{5}{4})})}}{2 \cdot (3^{-(\frac{3}{8}d + \frac{5}{4})} + 1)}.$$

By Lemma 6, $G[V(\mathcal{T})]$ is 11-degenerate and thus $d \leq 22$. Furthermore, using the $\frac{9}{4}$ -approximation algorithm [15] for planar bipartization, we can set $\rho_0 = \frac{9}{4}$. Plugging in these values to the above inequality, we have $\rho \leq 2.99993033741$.

Our entire algorithm first applies Lemma 4 with a maximal packing \mathcal{C} of $K_4(G)$ to reduce the problem to $G - V(\mathcal{C})$, which is a K_4 -free disk graph, and then applies Algorithm 1 on $G - V(\mathcal{C})$. By Lemma 4 and the above analysis, this algorithm solves BIPARTIZATION on disk graphs with an *expected* approximation ratio at most 2.99993033741. By repeating the algorithm polynomial number of times, we can also obtain a randomized algorithm that achieves the same approximation ratio with high probability.

► **Theorem 14.** *There exists a polynomial-time randomized algorithm for BIPARTIZATION on the class of disk graphs that gives a $(3 - \alpha)$ -approximation solution with high probability, for some $\alpha > 10^{-5}$.*

¹ In Lemma 12, the bound for $\mathbb{E}[|S_3|]$ has a condition $\rho \geq 2$. But we can assume this is always the case, for otherwise our algorithm is already a 2-approximation algorithm.

With some efforts, we can derandomize our algorithm to obtain a deterministic $(3 - \alpha)$ -approximation algorithm for BIPARTIZATION on disk graphs. In fact, in Algorithm 1, only the construction of the set R is randomized. We show in the full version how to construct R deterministically while still guaranteeing the nice properties of R (the key point is to guarantee that there are many dead vertices). The approximation ratio of our deterministic algorithm is slightly worse than the randomized one (while it is still smaller than 3).

► **Theorem 2.** *There exists a polynomial-time $(3 - \alpha)$ -approximation algorithm for BIPARTIZATION on the class of disk graphs, for some constant $\alpha > 0$.*

4 Generalizations

In this section, we discuss some generalizations of our result. First, we observe that our algorithm directly generalizes to *pseudo-disk graphs*. A set of geometric objects in the plane are called *pseudo-disks* if each of them is homeomorphic to a disk and the boundaries of any two objects intersect at most twice. A graph is a *pseudo-disk graph* if it can be represented as the intersection graph of a set of pseudo-disks.

In our BIPARTIZATION algorithm, we only exploit two properties of disk graphs: (i) triangle-free disk graphs are planar, and (ii) K_4 -free disk graphs are c -degenerate for some constant c . In fact, pseudo-disk graphs also satisfy these two properties.

► **Fact 15** ([22]). *Triangle-free pseudo-disk graphs are planar.*

► **Fact 16.** *K_4 -free pseudo-disk graphs are c -degenerate for some constant c .*

Proof. We show that any K_r -free pseudo-disk graph of n vertices only has $O(rn)$ edges, which implies the fact. Let G be a K_r -free pseudo-disk graph realized by a set \mathcal{S} of pseudo-disks. We say an edge (S, S') of G is an *inclusion edge* if $S \subseteq S'$ or $S' \subseteq S$. We first observe that G has $O(rn)$ inclusion edges. Indeed, a pseudo-disk $S \in \mathcal{S}$ cannot be contained in $r - 1$ (or more) other pseudo-disks in \mathcal{S} , for otherwise there is a copy of K_r in G . Thus, if we charge every inclusion edge (S, S') to the smaller one of S and S' , every pseudo-disk is charged at most $r - 2$ times. This implies that G has $O(rn)$ inclusion edges. Now we bound the number of other edges in G . Note that if (S, S') is a non-inclusion edge in G , then the boundaries of S and S' intersect. So it suffices to bound the total number of intersection points of the boundaries of the pseudo-disks in \mathcal{S} . The depth of an intersection point x is the number of pseudo-disks in \mathcal{S} containing x . It is well-known that in a set of n pseudo-disks, the number of boundary intersection points of depth at most d is bounded by $O(dn)$ [21]. Since G is K_r -free, every intersection point is of depth at most r . Thus, the total number of boundary intersection points is $O(rn)$, implying that G has $O(rn)$ edges. ◀

Therefore, our algorithm directly generalizes to pseudo-disk graphs.

► **Theorem 3.** *There exists a polynomial-time $(3 - \alpha)$ -approximation algorithm for BIPARTIZATION on the class of pseudo-disk graphs, for some constant $\alpha > 0$.*

Next, we observe that our techniques apply to not only the specific problem of BIPARTIZATION. In fact, it works for a wide class of vertex-deletion problems on (pseudo-)disk graphs. Recall that in a vertex-deletion problem, the goal is to delete a minimum set S of vertices from a graph G such that $G - S$ satisfies some desired property \mathbf{P} . In BIPARTIZATION, the property \mathbf{P} is “being bipartite”. Our technique applies to any vertex-deletion problem that is (i) hereditary, i.e., if a graph satisfies \mathbf{P} then all its induced subgraphs also satisfy \mathbf{P} , and (ii) triangle-conflicting, i.e., a graph satisfies \mathbf{P} only if it is triangle-free.

► **Theorem 17.** *If a hereditary and triangle-conflicting vertex-deletion problem admits a $(3 - \delta)$ -approximation algorithm on the class of planar graphs for some $\delta > 0$, then it admits a $(3 - \alpha)$ -approximation algorithm on the class of (pseudo-)disk graphs for some $\alpha > 0$.*

Proof. We just replace the PLANARBIP sub-routine in Algorithm 1 with the $(3 - \delta)$ -approximation algorithm for the problem on planar graphs. As one can easily verify, our analysis only depends on the fact that the vertex-deletion problem is hereditary and triangle-conflicting. Thus, the same analysis shows that this is a $(3 - \alpha)$ -approximation algorithm for the problem on (pseudo-)disk graphs. ◀

Well-studied instances of hereditary and triangle-conflicting vertex-deletion problems (other than BIPARTIZATION) include VERTEX COVER, FEEDBACK VERTEX SET, TRIANGLE HITTING, etc. Most of these problems already have $(3 - \delta)$ -approximation algorithms on disk graphs, and some of them do not have known $(3 - \delta)$ -approximation algorithms on planar graphs. Thus, at this point, we can only obtain interesting results for BIPARTIZATION. However, we believe that this is not the full power of Theorem 17. To provide some evidences, let us consider a vertex-deletion problem, PLANARIZATION&BIPARTIZATION, in which the desired property \mathbf{P} is “being planar and bipartite”. This problem is clearly hereditary and triangle-conflicting. On planar graphs, it is equivalent to BIPARTIZATION and thus admits a $(3 - \delta)$ -approximation algorithm. Therefore, Theorem 17 gives a $(3 - \alpha)$ -approximation algorithm for PLANARIZATION&BIPARTIZATION on (pseudo-)disk graphs. Although this problem itself is somehow artificial and not well-studied, it reveals that Theorem 17 could possibly have further applications in the future.

References

- 1 Amit Agarwal, Moses Charikar, Konstantin Makarychev, and Yury Makarychev. $O(\sqrt{\log n})$ approximation algorithms for min uncut, min 2cnf deletion, and directed cut problems. In Harold N. Gabow and Ronald Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 573–581. ACM, 2005. doi:10.1145/1060590.1060675.
- 2 Sayan Bandyapadhyay, William Lochet, Daniel Lokshtanov, Saket Saurabh, and Jie Xue. Subexponential parameterized algorithms for cut and cycle hitting problems on h-minor-free graphs. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2063–2084. SIAM, 2022.
- 3 Sayan Bandyapadhyay, William Lochet, Daniel Lokshtanov, Saket Saurabh, and Jie Xue. True contraction decomposition and almost ETH-tight bipartization for unit-disk graphs. In *38th International Symposium on Computational Geometry (SoCG 2022)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.
- 4 Nikhil Bansal and Subhash Khot. Optimal long code test with one free bit. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 453–462. IEEE, 2009.
- 5 Marthe Bonamy, Édouard Bonnet, Nicolas Bousquet, Pierre Charbit, Panos Giannopoulos, Eun Jung Kim, Pawel Rzazewski, Florian Sikora, and Stéphan Thomassé. EPTAS and subexponential algorithm for maximum clique on disk and unit ball graphs. *J. ACM*, 68(2):9:1–9:38, 2021. doi:10.1145/3433160.
- 6 Hyeon-Ah Choi, Kazuo Nakajima, and Chong S. Rim. Graph bipartization and via minimization. *SIAM J. Discret. Math.*, 2(1):38–47, 1989. doi:10.1137/0402004.
- 7 Mark de Berg, Hans L. Bodlaender, Sándor Kisfaludi-Bak, Dániel Marx, and Tom C. van der Zanden. A framework for exponential-time-hypothesis-tight algorithms and lower bounds in geometric intersection graphs. *SIAM J. Comput.*, 49(6):1291–1331, 2020. doi:10.1137/20M1320870.

- 8 Thomas Erlebach, Klaus Jansen, and Eike Seidel. Polynomial-time approximation schemes for geometric intersection graphs. *SIAM J. Comput.*, 34(6):1302–1323, 2005. doi:10.1137/S0097539702402676.
- 9 Guy Even, Joseph Naor, Satish Rao, and Baruch Schieber. Divide-and-conquer approximation algorithms via spreading metrics. *J. ACM*, 47(4):585–616, 2000. doi:10.1145/347476.347478.
- 10 Samuel Fiorini, Nadia Hardy, Bruce A. Reed, and Adrian Vetta. Approximate min-max relations for odd cycles in planar graphs. *Math. Program.*, 110(1):71–91, 2007. doi:10.1007/s10107-006-0063-7.
- 11 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Finding, hitting and packing cycles in subexponential time on unit disk graphs. *Discret. Comput. Geom.*, 62(4):879–911, 2019. doi:10.1007/s00454-018-00054-x.
- 12 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, and Meirav Zehavi. ETH-tight algorithms for long path and cycle on unit disk graphs. *J. Comput. Geom.*, 12(2):126–148, 2021. doi:10.20382/jocg.v12i2a6.
- 13 Matt Gibson and Imran A. Pirwani. Algorithms for dominating set in disk graphs: Breaking the $\log n$ barrier – (extended abstract). In Mark de Berg and Ulrich Meyer, editors, *Algorithms – ESA 2010, 18th Annual European Symposium, Liverpool, UK, September 6-8, 2010. Proceedings, Part I*, volume 6346 of *Lecture Notes in Computer Science*, pages 243–254. Springer, 2010. doi:10.1007/978-3-642-15775-2_21.
- 14 Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, 1995. doi:10.1145/227683.227684.
- 15 Michel X. Goemans and David P. Williamson. Primal-dual approximation algorithms for feedback problems in planar graphs. *Comb.*, 18(1):37–59, 1998. doi:10.1007/PL00009810.
- 16 Bart MP Jansen and Stefan Kratsch. On polynomial kernels for structural parameterizations of odd cycle transversal. In *International Symposium on Parameterized and Exact Computation*, pages 132–144. Springer, 2011.
- 17 Bart MP Jansen, Marcin L Pilipczuk, and Erik Jan Van Leeuwen. A deterministic polynomial kernel for odd cycle transversal and vertex multiway cut in planar graphs. *SIAM Journal on Discrete Mathematics*, 35(4):2387–2429, 2021.
- 18 Andrew B. Kahng, Shailesh Vaya, and Alexander Zelikovsky. New graph bipartizations for double-exposure, bright field alternating phase-shift mask layout. In Satoshi Goto, editor, *Proceedings of ASP-DAC 2001, Asia and South Pacific Design Automation Conference 2001, January 30-February 2, 2001, Yokohama, Japan*, pages 133–138. ACM, 2001. doi:10.1145/370155.370304.
- 19 Ken-Ichi Kawarabayashi and Atsuhiko Nakamoto. The Erdős–Pósa property for vertex-and edge-disjoint odd cycles in graphs on orientable surfaces. *Discrete Mathematics*, 307(6):764–768, 2007.
- 20 Ken-ichi Kawarabayashi and Bruce A. Reed. Odd cycle packing. In Leonard J. Schulman, editor, *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 695–704. ACM, 2010. doi:10.1145/1806689.1806785.
- 21 Klara Kedem, Ron Livne, János Pach, and Micha Sharir. On the union of Jordan regions and collision-free translational motion amidst polygonal obstacles. *Discrete & Computational Geometry*, 1(1):59–71, 1986.
- 22 Jan Kratochvíl. Intersection graphs of noncrossing arc-connected sets in the plane. In *International Symposium on Graph Drawing*, pages 257–270. Springer, 1996.
- 23 Stefan Kratsch and Magnus Wahlström. Compression via matroids: a randomized polynomial kernel for odd cycle transversal. *ACM Transactions on Algorithms (TALG)*, 10(4):1–15, 2014.
- 24 Stefan Kratsch and Magnus Wahlström. Representative sets and irrelevant vertices: New tools for kernelization. *J. ACM*, 67(3):16:1–16:50, 2020. doi:10.1145/3390887.

- 25 Daniel Lokshtanov, N. S. Narayanaswamy, Venkatesh Raman, M. S. Ramanujan, and Saket Saurabh. Faster parameterized algorithms using linear programming. *ACM Trans. Algorithms*, 11(2):15:1–15:31, 2014. doi:10.1145/2566616.
- 26 Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Jie Xue, and Meirav Zehavi. Subexponential parameterized algorithms on disk graphs (extended abstract). In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2005–2031. SIAM, 2022.
- 27 Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Jie Xue, and Meirav Zehavi. A 1.9999-approximation algorithm for vertex cover on string graphs. In *40th International Symposium on Computational Geometry (SoCG 2024)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024.
- 28 Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Jie Xue, and Meirav Zehavi. A framework for approximation schemes on disk graphs. In *34rd Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2023.
- 29 Daniel Lokshtanov, Saket Saurabh, and Magnus Wahlström. Subexponential parameterized odd cycle transversal on planar graphs. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2012)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2012.
- 30 Madhav V. Marathe, Heinz Breu, Harry B. Hunt, S. S. Ravi, and Daniel J. Rosenkrantz. Simple heuristics for unit disk graphs. *Networks*, 25:59–68, 1995.
- 31 Dániel Marx, Pranabendu Misra, Daniel Neuen, and Prafullkumar Tale. A framework for parameterized subexponential algorithms for generalized cycle hitting problems on planar graphs. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2085–2127. SIAM, 2022.
- 32 Mihai Pop, Daniel S Kosack, and Steven L Salzberg. Hierarchical scaffolding with bambus. *Genome research*, 14(1):149–159, 2004.
- 33 Dieter Rautenbach and Bruce Reed. The Erdos-Pósa property for odd cycles in highly connected graphs. *Combinatorica*, 21(2):267–278, 2001.
- 34 Dieter Rautenbach and Bruce A. Reed. The Erdos-Pósa property for odd cycles in highly connected graphs. *Comb.*, 21(2):267–278, 2001. doi:10.1007/s004930100024.
- 35 Bruce Reed, Kaleigh Smith, and Adrian Vetta. Finding odd cycle transversals. *Operations Research Letters*, 32(4):299–301, 2004.
- 36 Bruce A. Reed. Mangoes and blueberries. *Comb.*, 19(2):267–296, 1999. doi:10.1007/s004930050056.
- 37 Bruce A. Reed, Kaleigh Smith, and Adrian Vetta. Finding odd cycle transversals. *Oper. Res. Lett.*, 32(4):299–301, 2004. doi:10.1016/j.orl.2003.10.009.
- 38 Romeo Rizzi, Vineet Bafna, Sorin Istrail, and Giuseppe Lancia. Practical algorithms and fixed-parameter tractability for the single individual SNP haplotyping problem. In Roderic Guigó and Dan Gusfield, editors, *Algorithms in Bioinformatics, Second International Workshop, WABI 2002, Rome, Italy, September 17-21, 2002, Proceedings*, volume 2452 of *Lecture Notes in Computer Science*, pages 29–43. Springer, 2002. doi:10.1007/3-540-45784-4_3.
- 39 Sartaj Sahni and Teofilo F. Gonzalez. P-complete approximation problems. *J. ACM*, 23(3):555–565, 1976. doi:10.1145/321958.321975.
- 40 Carsten Thomassen. The Erdős-Pósa property for odd cycles in graphs of large connectivity. *Combinatorica*, 21(2):321–333, 2001.
- 41 Erik Jan van Leeuwen. Better approximation schemes for disk graphs. In Lars Arge and Rusins Freivalds, editors, *Algorithm Theory – SWAT 2006, 10th Scandinavian Workshop on Algorithm Theory, Riga, Latvia, July 6-8, 2006, Proceedings*, volume 4059 of *Lecture Notes in Computer Science*, pages 316–327. Springer, 2006. doi:10.1007/11785293_30.
- 42 Sebastian Wernicke. *On the algorithmic tractability of single nucleotide polymorphism (SNP) analysis and related problems*. diplom. de, 2014.