True Contraction Decomposition and Almost ETH-Tight Bipartization for Unit-Disk Graphs

Sayan Bandyapadhyay ✉
University of Bergen, Norway

William Lochet ✉
LIRMM, Université de Montpellier, CNRS, France

Daniel Lokshtanov ✉
University of California, Santa Barbara, CA, USA

Saket Saurabh ✉
Institute of Mathematical Sciences, Chennai, India

Jie Xue ✉
New York University Shanghai, China

Abstract

We prove a structural theorem for unit-disk graphs, which (roughly) states that given a set $\mathcal{D}$ of n unit disks inducing a unit-disk graph $G_{\mathcal{D}}$ and a number $p \in [n]$, one can partition $\mathcal{D}$ into p subsets $\mathcal{D}_1, \ldots, \mathcal{D}_p$ such that for every $i \in [p]$ and every $\mathcal{D}' \subseteq \mathcal{D}_i$, the graph obtained from $G_{\mathcal{D}}$ by contracting all edges between the vertices in $\mathcal{D}_i \backslash \mathcal{D}'$ admits a tree decomposition in which each bag consists of $O(p + |\mathcal{D}'|)$ cliques. Our theorem can be viewed as an analog for unit-disk graphs of the structural theorems for planar graphs and almost-embeddable graphs proved very recently by Marx et al. [SODA'22] and Bandyapadhyay et al. [SODA'22].

By applying our structural theorem, we give several new combinatorial and algorithmic results for unit-disk graphs. On the combinatorial side, we obtain the first Contraction Decomposition Theorem (CDT) for unit-disk graphs, resolving an open question in the work Panolan et al. [SODA'19]. On the algorithmic side, we obtain a new FPT algorithm for bipartization (also known as odd cycle transversal) on unit-disk graphs, which runs in $2^{O(\sqrt{k} \log k)} \cdot n^{O(1)}$ time, where k denotes the solution size. Our algorithm significantly improves the previous slightly subexponential-time algorithm given by Lokshtanov et al. [SODA'22] (which works more generally for disk graphs) and is almost optimal, as the problem cannot be solved in $2^{o(\sqrt{k})} \cdot n^{O(1)}$ time assuming the ETH.

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

Keywords and phrases unit-disk graphs, tree decomposition, contraction decomposition, bipartization

Digital Object Identifier 10.4230/LIPIcs.SoCG.2022.11

Funding *Daniel Lokshtanov*: Supported by BSF award 2018302 and NSF award CCF-2008838.
Saket Saurabh: Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819416), and Swarnajayanti Fellowship (No. DST/SJF/MSA01/2017-18).

1 Introduction

For a set $\mathcal{D}$ of unit disks in the plane, the unit-disk graph $G_{\mathcal{D}}$ induced by $\mathcal{D}$ has the unit disks in $\mathcal{D}$ as its vertices, where two vertices are connected by an edge whenever the two corresponding unit disks intersect. As one of the simplest but most important classes of geometric intersection graphs, unit-disk graphs have been extensively studied in various areas (e.g., computational geometry, graph theory, algorithms) and find applications such as modeling the topology of ad-hoc communication networks [27, 49]. The research on unit-disk graphs focused on both combinatorial aspects and algorithmic aspects.

In this paper, we establish a structural theorem for unit-disk graphs, which leads to interesting new results in both combinatorial and algorithmic aspects. Our theorem can be viewed as a unit-disk-graph analog of the very recent theorems proved for planar graphs [39] and more generally for the so-called "almost-embeddable" graphs [5]. Thus, before introducing our theorem, let us first briefly review their results. Specifically, it was shown in [5, 39] that for a planar graph $G = (V, E)$ and a number $p \in [n]$ where $n = |V|$, one can partition V into $V_1, \dots, V_p$ such that for every $i \in [p]$ and $V' \subseteq V_i$, the graph obtained from G by contracting all edges between the vertices in $V_i \backslash V'$ has treewidth $O(p + |V'|)$. Unfortunately, one can easily see that such a statement cannot hold for unit-disk graphs[1]. However, if we use the number of *cliques* (instead of vertices) in the bags of the tree decomposition to define its width, this statement is actually true for unit-disk graphs!

Let $\mathcal{D}$ be a set of n unit disks and $p \in [n]$ be any number. Our theorem (roughly) states that one can partition $\mathcal{D}$ into p subsets $\mathcal{D}_1, \dots, \mathcal{D}_p$ such that for every $i \in [p]$ and every $\mathcal{D}' \subseteq \mathcal{D}_i$, the graph obtained from the unit-disk graph $G_{\mathcal{D}}$ by contracting all edges between the vertices in $\mathcal{D}_i \backslash \mathcal{D}'$ admits a tree decomposition in which each bag consists of $O(p + |\mathcal{D}'|)$ cliques. Furthermore, this partition can be computed in polynomial time. The formal statement of our theorem is more technical, and will be presented in Theorem 2 after we introduce some preliminaries in Section 2. Note that the notion of tree decomposition with bags consisting of cliques is not new. In fact, this kind of tree decomposition has been widely applied on unit-disk graphs and other geometric intersection graphs to design efficient algorithms; see for example [12, 21, 43]. In what follows, we discuss the new combinatorial and algorithmic results derived from our main theorem.

Combinatorial application: the first CDT on unit-disk graphs. In graph theory, a *Contraction Decomposition Theorem* (CDT) is a statement of the following form: given a graph $G = (V, E)$ from some graph class, for any $p \in \mathbb{N}$, one can partition E into $E_1, \dots, E_p$ such that contracting the edges in each E_i in G yields a graph of treewidth at most $f(p)$, for some function $f : \mathbb{N} \to \mathbb{N}$. CDT is classical tool useful in designing efficient approximation and parameterized algorithms in certain classes of graphs. Graph classes for which CDTs are known include planar graphs [31, 32], graphs of bounded genus [15], and H-minor free graphs [14]. However, little was known about CDTs on geometric intersection graphs. Recently, Panolan et al. [44] made the first progress towards proving a CDT for unit-disk graphs. Specifically, they gave a weak version of CDT (which they call a *relaxed* CDT), in which the edge sets $E_1, \dots, E_p$ need not to be disjoint; instead, it is required that each edge $e \in E$ is contained in $O(1)$ sets in $E_1, \dots, E_p$. It remains open whether unit-disk graphs admit a "true" CDT (where $E_1, \dots, E_p$ is a partition of E). In this paper, by applying our main theorem, we give the first CDT for unit-disk graphs and hence resolve an open question of [44] (and also Hajiaghayi [26]). The function f in our CDT is quadratic, i.e., $f(p) = O(p^2)$, matching the bound in the weak CDT of [44].

Algorithmic application: almost ETH-tight bipartization on unit-disk graphs. Designing efficient algorithms on unit-disk graphs is a central topic in the study of unit-disk graphs. Many classical algorithmic problems have been studied on unit-disk graphs. Polynomial-time solvable problems include shortest paths [7, 8, 47], diameter computing [9, 24], maximum clique [10], etc. Compared to these problems, NP-hard problems received more attentions

[1] Indeed, the clique K_n is a unit-disk graph, and if we partition the vertices of K_n into p parts for $p \geq 2$, after contracting the smallest part, we get a clique of size at least $n/2$ which has treewidth $\Omega(n)$.

on unit-disk graphs. In particular, studying parametrized algorithms [11] for these hard problems on unit-disk graphs (or other geometric intersection graphs) is one of the most active themes in recent years [2, 3, 20, 21, 22, 23, 43] (also see the survey [44]). A well-known fact about parametrized complexity on planar graphs (or more generally, bounded-genus graphs and H-minor-free graphs) is the so-called "square root phenomenon": many problems on planar graphs admit algorithms with running time $2^{\widetilde{O}(\sqrt{k})}n^{O(1)}$ or $n^{\widetilde{O}(\sqrt{k})}$, where k is the parameter (usually the solution size), and also admit (almost) matching lower bounds [6, 13, 16, 18, 19, 33, 34, 40, 42, 46]. Recently, it was shown that such a "square root phenomenon" also appears in many problems on unit disk graphs. Specifically, algorithms with running time $2^{\widetilde{O}(\sqrt{k})}n^{O(1)}$ or $n^{\widetilde{O}(\sqrt{k})}$ were obtained on unit-disk graphs for Vertex Cover [12], Independent Set [41], Feedback Vertex Set [4, 20], k-Path/Cycle [20, 22], etc. and (almost) matching lower bounds were also known [12]. In this paper, we apply our main theorem to add another classical problem to this family, namely, Bipartization.

In the Bipartization problem, one aims to make a graph bipartite by deleting few vertices. Formally, the input of Bipartization is a graph $G = (V, E)$ and a number k, and the goal is to determine whether there exists $X \subseteq V$ of size at most k such that $G - X$ is *bipartite*. In the literature, Bipartization is also called Odd Cycle Transversal, as making a graph bipartite is equivalent to hitting all its odd cycles. As one of the most fundamental NP-complete problems in graph theory [48], Bipartization has been studied extensively over years [1, 17, 25, 28, 29, 30, 35, 45]. The best existing algorithm for Bipartization on general graphs runs in $2.3146^k n^{O(1)}$ time [36]. On planar graphs, a randomized algorithm with running time $2^{O(\sqrt{k}\log k)}n^{O(1)}$ was known [38, 39], and the same running time was achieved also for bounded-genus graphs and H-minor-free graphs very recently [5]. However, little was known about Bipartization on geometric intersection graphs. In fact, even achieving *slightly* subexponential-time parameterized algorithm for Bipartization on unit-disk graphs was a long-standing open problem, prior to the very recent work by Lokshtanov et al. [37]. The authors of [37] gave a randomized algorithm running in $2^{O(k^{\frac{27}{28}}\log k)}n^{O(1)}$ time for Bipartization on disk graphs (and thus unit-disk graphs), achieving the first $2^{o(k)}$ bound for the problem. This result, however, is still far away from showing the "square root phenomenon" for Bipartization on unit-disk graphs.

By applying our main theorem, we solve Bipartization on unit-disk graphs with a randomized algorithm running in $2^{O(\sqrt{k}\log k)}n^{O(1)}$ time, significantly improving the $2^{O(k^{\frac{27}{28}}\log k)}$ bound given by [37]. On the other hand, we establish an almost matching lower bound, showing that the problem cannot be solved in in $2^{o(\sqrt{k})}n^{O(1)}$ time, assuming the Exponential Time Hypothesis (ETH). Our results thus add Bipartization to the "square root" family of problems on unit-disk graphs. In terms of techniques, our algorithm solves the problem by first constructing the partition $\{\mathcal{D}_1, \dots, \mathcal{D}_p\}$ of the unit-disk set $\mathcal{D}$ in our main theorem for $p = \sqrt{k}$ and then applying the well-known Baker's technique on $\mathcal{D}_1, \dots, \mathcal{D}_p$ together with a DP procedure similar to the one in [5] on tree decomposition. Such a scheme based on our theorem can possibly also be applied to solve other problems on unit-disk graphs. To give an example, we extend our algorithm to the problem of Group Feedback Vertex Set with non-identity labels, with the same running time.

Due to limited space, some proofs/details are omitted in this version, and will appear in the full paper.

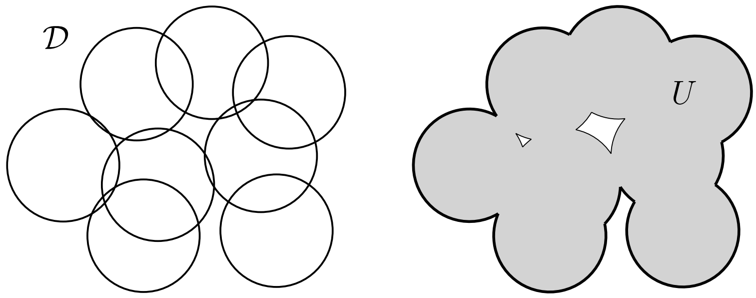

Figure 1 The boundary and outer boundary of U (the heavier curve is the outer boundary).

2 Preliminaries

The canonical grid. Consider the grid formed by vertical lines $\{x = i : i \in \mathbb{N}\}$ and horizontal lines $\{y = i : i \in \mathbb{N}\}$. We shall use it as the *canonical* grid throughout this paper (in the rest of the paper, we shall refer it as "the grid"). Each cell in the grid is a unit square, and we usually use the notation $\Box$ to denote a cell. For a unit disk D, we denote by $\Box_D$ the grid cell that contains the center of D. For a set $\mathcal{D}$ of unit disks and a cell $\Box$, we denote by $\mathcal{D} \Cap \Box$ the subset of unit disks in $\mathcal{D}$ whose centers lie in $\Box$. We say a subset $\mathcal{D}' \subseteq \mathcal{D}$ is *grid-respecting* if for any cell $\Box$ such that $\mathcal{D}' \Cap \Box \neq \emptyset$, we have $\mathcal{D}' \Cap \Box = \mathcal{D} \Cap \Box$. A partition $\{\mathcal{D}_1, \ldots, \mathcal{D}_p\}$ of $\mathcal{D}$ is *grid-respecting* if $\mathcal{D}_1, \ldots, \mathcal{D}_p$ are all grid-respecting subsets of $\mathcal{D}$.

Basic graph notions. Let $G = (V, E)$ be a graph. For a subset $V' \subseteq V$, the *subgraph* of G *induced* by V' is the graph consisting of the vertices in V' and the edges in E with both endpoints in V'. An *induced subgraph* of G is a subgraph of G induced by a subset of V. A vertex $v \in V$ is *neighboring* to a subset $V' \subseteq V$ in G if there exists $v' \in V'$ such that $(v, v') \in E$. A subset $V' \subseteq V$ is *neighboring* to another subset $V'' \subseteq V$ if there exist $v' \in V'$ and $v'' \in V''$ such that $(v', v'') \in E$.

Unit disks and unit-disk graphs. Let $\mathcal{D}$ be a set of unit disks in the plane. For $D \in \mathcal{D}$, we denote by $\mathsf{ctr}(D)$ the *center* of the unit disk D. The union $U = \bigcup_{D \in \mathcal{D}} D$ is a closed region in the plane, whose boundary consists of a set of disjoint closed curves. The *outer boundary* of U is defined as the part of the boundary of U that is incident to the unbounded connected component of $\mathbb{R}^2 \backslash U$; see Figure 1 for an illustration. The *exposed* unit disks in $\mathcal{D}$ refers to the unit disks in $\mathcal{D}$ that intersect the outer boundary of U. In Figure 1, all unit disks in $\mathcal{D}$ are exposed. We denote by $\mathsf{Exp}(\mathcal{D})$ the set of exposed unit disks in $\mathcal{D}$. The *unit-disk graph* induced by $\mathcal{D}$, denoted by $G_{\mathcal{D}}$, has the unit disks in $\mathcal{D}$ as its vertices, where two vertices are connected by an edge whenever the two corresponding unit disks intersect. We use $E_{\mathcal{D}}$ to denote the edge set of $G_{\mathcal{D}}$. Note that for a cell $\Box$, the unit disks in $\mathcal{D} \cap \Box$ pairwise intersect and hence form a clique in $G_{\mathcal{D}}$, which we call a *cell clique*. We denote by $E^*_{\mathcal{D}} \subseteq E_{\mathcal{D}}$ the set of edges in all cell cliques in $G_{\mathcal{D}}$. For a subset $\mathcal{D}' \subseteq \mathcal{D}$, the unit-disk graph $G_{\mathcal{D}'}$ is canonically isomorphic to the subgraph of $G_{\mathcal{D}}$ induced by $\mathcal{D}'$. Thus, for convenience, we shall not distinguish between them: we shall also use $G_{\mathcal{D}'}$ to denote the induced subgraph of $G_{\mathcal{D}}$ and use $E_{\mathcal{D}'}$ to denote the set of edges in $G_{\mathcal{D}}$ between the vertices in $\mathcal{D}'$.

Tree decomposition and treewidth. A *tree decomposition* of a graph $G = (V, E)$ is a pair (T, β) where T is a tree and $\beta : T \to 2^V$ maps the nodes of T to subsets of V such that **(i)** $\bigcup_{t \in T} \beta(t) = V$, **(ii)** for each edge $(u, v) \in E$, there exists $t \in T$ with $u, v \in \beta(t)$, and **(iii)** for each vertex $v \in V$, the nodes $t \in T$ with $v \in \beta(t)$ form a connected subset in T. Conventionally, we call $\beta(t)$ the *bag* of the node $t \in T$. The *width* of the tree decomposition (T, β) is $\max_{t \in T} |\beta(t)| - 1$. The *treewidth* of a graph G, denoted by $\mathbf{tw}(G)$ is the minimum width of a tree decomposition of G. It is sometimes more convenient to consider *rooted* trees. Thus, throughout this paper, we always view the tree in a tree decomposition as a rooted tree. A tree decomposition (T, β) is *binary* if T is binary.

Edge contraction. From a graph $G = (V, E)$, one can obtain a new graph via a so-called *edge contraction* operation. Specifically, by contracting an edge $e = (u, v) \in E$, we merge u and v into one vertex with edges connecting to both the neighbors of u and the neighbors of v in $V \backslash \{u, v\}$. More generally, we can contract a subset $E_0 \subseteq E$ of edges simply by contracting these edges "one-by-one". Formally, the resulting graph by contracting E_0 in G, which we denote by G/E_0, is defined as follows. The vertices of G/E_0 one-to-one corresponds to the connected components of the graph $G_0 = (V, E_0)$, and two vertices have an edge connecting them whenever the corresponding two connected components of G_0 are neighboring in G. Let V_0 denote the vertex set of G/E_0. Associated to this edge contraction, there is a natural map $\pi : V \to V_0$ which maps each vertex $v \in V$ to the vertex of G/E_0 corresponding to the connected component of G_0 that contains v. We call π the *quotient map* of the edge contraction. The following fact is a well-known (and can be easily verified).

▶ **Fact 1.** *Let $G = (V, E)$ be a graph obtained from another graph $G' = (V', E')$ via edge contraction with quotient map $\pi : V' \to V$. The following statements are true.*

(i) *If (T, β) is a tree decomposition of G, then (T, β') is a tree decomposition of G' where $\beta'(t) = \pi^{-1}(\beta(t))$ for all nodes $t \in T$.*

(ii) *If (T', β') is a tree decomposition of G', then (T', β) is a tree decomposition of G where $\beta(t) = \pi(\beta'(t))$ for all nodes $t \in T'$.*

3 The main theorem

In this section, we present the main theorem of this paper, which establishes a structural property of unit-disk graphs. Formally, the theorem is the following.

▶ **Theorem 2** (main theorem)**.** *Given a set $\mathcal{D}$ of n unit disks and an integer $p \in [n]$, one can compute in polynomial time a grid-respecting partition $\{\mathcal{D}_1, \dots, \mathcal{D}_p\}$ of $\mathcal{D}$ such that for every $i \in [p]$ and every $\mathcal{D}' \subseteq \mathcal{D}_i$, $\mathbf{tw}(G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i \backslash \mathcal{D}'})) = O(p + |\mathcal{D}'|)$.*

Recall that in Section 1, we gave an informal version of the above theorem, which states that $G_{\mathcal{D}}/E_{\mathcal{D}_i \backslash \mathcal{D}'}$ admits a tree decomposition in which each bag contains $O(p + |\mathcal{D}'|)$ cliques. One may ask how Theorem 2 implies this statement. To see this, observe that $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i \backslash \mathcal{D}'})$ can be viewed as a graph obtained from $G_{\mathcal{D}}/E_{\mathcal{D}_i \backslash \mathcal{D}'}$ via edge contraction. Thus, if we start from a tree decomposition of $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i \backslash \mathcal{D}'})$ of width $O(p + |\mathcal{D}'|)$ and apply Fact 1 to obtain a tree decomposition of $G_{\mathcal{D}}/E_{\mathcal{D}_i \backslash \mathcal{D}'}$, one can check that each bag of the latter tree decomposition consists of $O(p + |\mathcal{D}'|)$ cliques. We omit the details of this argument as it is not important. The rest of this section is dedicated to proving Theorem 2.

3.1 A layering for the unit disks

The first step of proving Theorem 2 is to compute a *layering* for the unit disks in $\mathcal{D}$, that is, a decomposition of $\mathcal{D}$ into *layers*. We shall use a function $\ell : \mathcal{D} \to \mathbb{N}$ to represent the layering: the unit disks which are mapped to i by ℓ form the i-th layer of $\mathcal{D}$. This layering ℓ *respects* the grid partition of $\mathcal{D}$ in the sense that $\ell^{-1}(\{i\})$ is a grid-respecting subset of $\mathcal{D}$ for all $i \in \mathbb{N}$. Besides, ℓ possesses some nice properties which will be used later to prove Theorem 2. Algorithm 1 presents the procedure for computing ℓ. In words, it iteratively finds the exposed unit disks in $\mathcal{D}$ (line 4) and removes from $\mathcal{D}$ the unit disks whose centers lie in the same cells as the centers of the exposed ones (line 5 and 7), and finally combines the unit disks removed in every 100 iterations into one layer (line 8).

Algorithm 1 LAYERING($\mathcal{D}$). ▷ Output a layering $\ell : \mathcal{D} \to \mathbb{N}$.

1: $q \leftarrow 0$
2: **while** $\mathcal{D} \neq \emptyset$ **do**
3: $\quad q \leftarrow q + 1$
4: $\quad \mathcal{X} \leftarrow \mathsf{Exp}(\mathcal{D})$
5: $\quad \mathcal{X}^+ = \bigcup_{X \in \mathcal{X}} (\mathcal{D} \Cap \Box_X)$
6: $\quad \mathsf{Tag}_X \leftarrow q$ for all $X \in \mathcal{X}^+$
7: $\quad \mathcal{D} \leftarrow \mathcal{D} \backslash \mathcal{X}^+$
8: **return** $\ell : D \mapsto \lceil \mathsf{Tag}_D / 100 \rceil$

It is clear that the layering ℓ returned by Algorithm 1 respects the cell partition of $\mathcal{D}$, because in line 6 we always assign the same tag to all unit disks with centers in the cells $\Box_D$. We write $\mathcal{L}_i = \ell^{-1}(\{i\})$ and call it *the i-th layer* of $\mathcal{D}$. Suppose there are in total m layers. We define $\mathcal{L}_{>i} = \bigcup_{j=i+1}^{m} \mathcal{L}_j$, $\mathcal{L}_{\geq i} = \bigcup_{j=i}^{m} \mathcal{L}_j$, $\mathcal{L}_{<i} = \bigcup_{j=1}^{i-1} \mathcal{L}_j$, $\mathcal{L}_{\leq i} = \bigcup_{j=1}^{i} \mathcal{L}_j$, and $\mathcal{L}_{[i,i']} = \bigcup_{j=i}^{i'} \mathcal{L}_j$. Next, we establish some nice properties of the layering ℓ.

▶ **Lemma 3.** *The layering ℓ and the layers $\mathcal{L}_1, \dots, \mathcal{L}_m$ satisfy the following three properties.*

(i) *For any $D, D' \in \mathcal{D}$ such that $D \cap D' \neq \emptyset$, we have $|\ell(D) - \ell(D')| \leq 1$.*

(ii) *For a connected component of $G_{\mathcal{L}_{>i}}$ with vertex set $\mathcal{C} \subseteq \mathcal{L}_{>i}$, the unit disks in $\mathcal{L}_i$ neighboring to $\mathcal{C}$ lie in the same connected component of $G_{\mathcal{L}_i}$.*

(iii) *For any $i, i' \in [m]$ with $i \leq i'$, $\mathbf{tw}\left(G_{\mathcal{L}_{[i,i']}} \middle/ E^*_{\mathcal{L}_{[i,i']}}\right) = O(i' - i + 1)$.*

We remark that the construction of our layering ℓ on unit-disk graphs is analogous to (and also inspired by) the outerplanarity layering on planar graphs (which is obtained by iteratively removing the vertices on the boundary of the outer face of the planar graph). While for the outerplanarity layering the three properties in Lemma 3 follow easily, it requires considerably more work to show them for our layering on unit-disk graphs.

In the rest of this section, we prove Lemma 3. We begin with introducing some notations for ease of exposition. Since $\mathcal{D}$ changes during Algorithm 1, we denote by $\mathcal{D}^{(q)}$ the set $\mathcal{D}$ at the beginning of the q-th iteration of the while-loop (line 2-7). Define $\mathcal{X}^{(q)} = \mathsf{Exp}(\mathcal{D}^{(q)})$ and $U^{(q)}$ as the union of the unit disks in $\mathcal{D}^{(q)}$.

Verifying property (i). Let $D, D' \in \mathcal{D}$ such that $D \cap D' \neq \emptyset$. To show $|\ell(D) - \ell(D')| \leq 1$, it suffices to show $|\mathsf{Tag}_D - \mathsf{Tag}_{D'}| \leq 100$. Let $q = \mathsf{Tag}_D$ and $q' = \mathsf{Tag}_{D'}$. If $q = q'$, we are done. If $q \neq q'$, we may assume $q < q'$ without loss of generality. Since $\mathsf{Tag}_D = q$, $D \in \mathcal{D} \Cap \Box_X$ for some $X \in \mathcal{X}^{(q)}$. By the definition of $\mathcal{X}^{(q)}$, X intersects the outer boundary of $U^{(q)}$ and thus there exists a point $x \in X$ that is on the outer boundary of $U^{(q)}$. Let σ be the segment connecting x and $d' = \mathsf{ctr}(D')$. We say a cell $\Box$ is *relevant* if there exists a unit disk in $\mathcal{D} \Cap \Box$ that intersects σ. We observe that there are at least $q' - q + 1$ relevant cells.

▶ **Observation 4.** *For each $i \in \{q, \ldots, q'\}$, there exists a unit disk $D_i \in \mathcal{D}$ with $\mathsf{Tag}_{D_i} = i$ that intersects σ. Thus, the number of relevant cells is at least $q' - q + 1$.*

Note that the length of σ is at most 3 because $D \cap D' \neq \emptyset$ and $D \cap X \neq \emptyset$. As such, there can be no more than 100 relevant cells (actually much fewer), because each relevant cell must contain a point with distance at most 1 from σ. Thus, $q' - q + 1 \leq 100$ and $|\ell(D) - \ell(D')| \leq 1$. Property (i) in Lemma 3 holds.

Verifying property (ii). Consider a connected component of $G_{\mathcal{L}_{>i}}$ with vertex set $\mathcal{C} \subseteq \mathcal{L}_{>i}$. Define $Q = \{q : \lceil q/100 \rceil = i\}$. For a fixed $q \in Q$, the outer boundary of $\mathcal{D}^{(q)}$ consists of some closed curves in the plane, each of which encloses a *region* that is topologically homeomorphic to a disk. These regions are clearly disjoint; we call the union of these regions the *domain* of $\mathcal{D}^{(q)}$. We claim that one of these regions should contain all unit disks in $\mathcal{C}$. First, observe that the domain of $\mathcal{D}^{(q)}$ contains all unit disks in $\mathcal{D}^{(q)}$, and hence contains all disks in $\mathcal{C}$ since $\mathcal{C} \subseteq \mathcal{L}_{>i} = \mathcal{D}^{(100i+1)} \subseteq \mathcal{D}^{(q)}$. Furthermore, because the regions are disjoint but $G_{\mathcal{C}}$ is connected, all unit disks in $\mathcal{C}$ must lie in the same region. We denote by R_q the region that contains the unit disks in $\mathcal{C}$. We do this for all $q \in Q$, and thus obtain a set $\{R_q\}_{q \in Q}$ of regions. We observe that these regions are nested.

▶ **Observation 5.** *$R_q \subseteq R_{q'}$ for all $q, q' \in Q$ with $q \geq q'$.*

To prove property (ii), consider two unit disks $D, D' \in \mathcal{L}_i$ that are neighboring to $\mathcal{C}$. Let $q = \mathsf{Tag}_D$ (resp., $q' = \mathsf{Tag}_{D'}$), then the tag of any unit disk in $\mathcal{D} \Cap \Box_D$ (resp., $\mathcal{D} \Cap \Box_{D'}$) is q (resp., q'). As $D, D' \in \mathcal{L}_i$, we have $q, q' \in Q$ and we assume $q \geq q'$ without loss of generality. Since D is neighboring to $\mathcal{C}$ and $\mathsf{Tag}_D = q$, D must be contained in R_q and thus all unit disks in $\mathcal{D} \Cap \Box_D$ are contained in R_q. Furthermore, there exists a unit disk $X \in \mathcal{D} \Cap \Box_D$ which is exposed in $\mathcal{D}^{(q)}$, i.e., $X \in \mathcal{X}^{(q)}$. Note that X must intersect the boundary of R_q, because X intersects the outer boundary of $U^{(q)}$ and is contained in R_q. Similarly, there exists a unit disk $X' \in \mathcal{D} \Cap \Box_{D'}$ exposed in $\mathcal{D}^{(q')}$ which intersects the boundary of $R_{q'}$.

▶ **Observation 6.** *$D' \cup X'$ intersects the boundary of R_q.*

Now both $D \cup X$ and $D' \cup X'$ are connected and intersect the boundary of R_q. Note that the unit disks in $\mathcal{D}^{(q)}$ that intersect the boundary of R_q form a connected unit-disk graph. Thus, the unit-disk graph induced by these unit disks together with D, X, D', X' is also connected. All these unit disks belong to $\mathcal{L}_i$, and are hence in the same connected component of $G_{\mathcal{L}_i}$. In particular, D and D' are in the same connected component of $G_{\mathcal{L}_i}$. Property (ii) in Lemma 3 holds.

Verifying property (iii). We notice that, in order to verify property (iii), it suffices to show that $\mathbf{tw}(G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}) = O(j)$ for all $j \in [m]$, because $\mathcal{L}_{[i,i']}$ is nothing but the first $j = i' - i + 1$ layers of the unit-disk set $\mathcal{L}_{\geq i}$. To this end, we first construct a drawing of the graph $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$ on the plane (possibly with edge crossings). The vertices of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$ one-to-one correspond to the cells $\Box$ for which $\mathcal{L}_{\leq j} \Cap \Box \neq \emptyset$, and we denote by $v(\Box)$ the vertex corresponding to the cell $\Box$. We draw each vertex $v(\Box)$ at an arbitrary point inside the cell $\Box$ that lies in the intersection of all unit disks in $\mathcal{D} \Cap \Box$ (such a point always exists, e.g., the center of $\Box$). For simplicity, we also use $v(\Box)$ to denote the point in the plane where we draw the vertex $v(\Box)$. For each edge $e = (v(\Box), v(\Box'))$ of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$, we draw it as a polyline (or polygonal chain) in the plane connecting $v(\Box)$ and $v(\Box')$ as follows. Since $v(\Box)$ and $v(\Box')$ are connected by an edge in $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$, there exist unit disks $D \in \mathcal{L}_{\leq j} \Cap \Box$ and $D' \in \mathcal{L}_{\leq j} \Cap \Box'$ such that $D \cap D' \neq \emptyset$. We choose an arbitrary point $x \in D \cap D'$ and let σ be the segment connecting x and $v(\Box)$, and σ' be the segment connecting x and $v(\Box')$. We then draw the edge e as the two-piece polyline consisting of the segments σ and σ', and denote this polyline by γ_e. See the left part of Figure 2 for an illustration. Note that γ_e is contained in $D \cup D'$. In this way, we obtain a plane drawing of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$ (possibly with edge crossings), and denote this drawing by η. For convenience, we call the polylines γ_e *edge curves*. Let $\varGamma$ be the image of η in the plane, which is equal to the union of all edge curves and all $v(\Box)$; see the right part of Figure 2. By our construction, $\varGamma$ is contained in the union of all unit disks in $\mathcal{D}$. Next, we establish some properties of $\varGamma$, which will be used later for bounding $\mathbf{tw}(G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}})$. For two points $a, b \in \mathbb{R}^2$, a *path* from a to b is a continuous map $f : [0, 1] \to \mathbb{R}^2$ with $f(0) = a$ and $f(1) = b$. We write $\Delta(f, \varGamma) = |\{x \in [0, 1] : f(x) \in \varGamma\}|$; if $\{x \in [0, 1] : f(x) \in \varGamma\}$ is not finite, we simply set $\Delta(f, \varGamma) = \infty$.

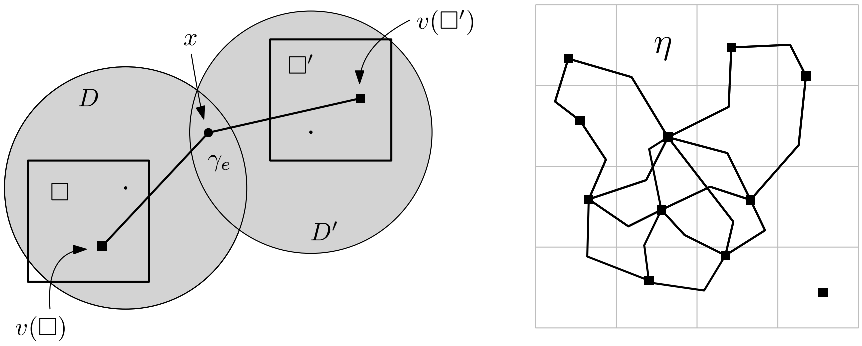

Figure 2 Illustrating the drawing η. The left part is the construction of one edge curve η_e and the right part is an example of how the drawing η finally looks like.

▶ **Observation 7.** *For any two points $a, b \in \mathbb{R}^2$ with distance $O(1)$, there exists a path $f : [0, 1] \to \mathbb{R}^2$ from a to b such that $\Delta(f, \varGamma) = O(1)$.*

▶ **Observation 8.** *For any point $a \in \mathbb{R}^2$, there exists a point b in the unbounded connected component of $\mathbb{R}^2 \backslash \varGamma$ and a path $f : [0, 1] \to \mathbb{R}^2$ from a to b such that $\Delta(f, \varGamma) = O(j)$.*

Proof sketch. We sketch a quick proof of this observation. First, by using Observation 7, one can easily see that for any point $a \in \mathbb{R}^2$, there is a path f from a to a vertex $v(\Box)$ of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$ with $\Delta(f, \varGamma) = O(1)$. So it suffices to consider the case where $a = v(\Box)$ for some vertex $v(\Box)$ of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$. Let q be the tag of the unit disks in $\mathcal{D} \Cap \Box$. We show the existence of a path f with $\Delta(f, \varGamma) = O(1)$ from $v(\Box)$ to some other vertex $v(\Box')$ such that the tag of the unit disks in $\mathcal{D} \Cap \Box'$ is smaller than q. Combining this with a simple induction argument completes the proof of the lemma. There are two cases: there exists such a vertex $v(\Box')$ with distance $O(1)$ from $v(\Box)$ or there does not exist. In the former case, we directly apply Observation 7 to obtain the path f from $v(\Box)$ to $v(\Box')$ with $\Delta(f, \varGamma) = O(1)$. In the latter case, we know there is no unit disk in $\mathcal{D} \backslash \mathcal{D}^{(q)}$ that is "close" to $v(\Box)$. However, some unit disk in $D \Cap \Box$ is exposed in $\mathcal{D}^{(q)}$ but not $\mathcal{D}^{(q-1)}$. That means $v(\Box)$ is close to a bounded connected component C of $\mathbb{R}^2 \backslash \bigcup_{D \in \mathcal{D}} D$, which is contained in the unbounded connected component of $\mathbb{R}^2 \backslash U^{(q)}$. In this case, we must have another vertex $v(\Box')$ close to C such that

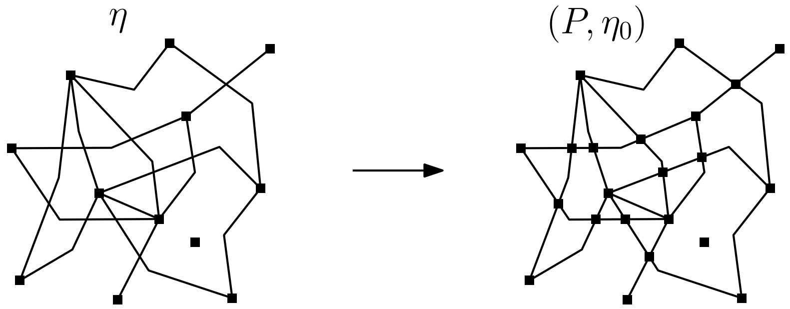

Figure 3 Illustrating the planar graph P obtained by adding vertices to the crossings of η.

the tag of the unit disks in $\mathcal{D} \Cap \Box'$ is smaller than q. We then construct the path f from $v(\Box)$ to $v(\Box')$ by first moving from $v(\Box)$ into C, then moving inside C to get close to $v(\Box')$, and finally moving out from C to $v(\Box')$. This summarizes the basic idea of the proof (though the complete proof is more complicated). ◀

The plane drawing η of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$ naturally induces a planar graph P as follows. The vertex set of P consists of the vertices of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$ and the edge-crossing points in the drawing η (called *crossings* for short). Two vertices of P are connected by an edge if they are "adjacent" on some edge curve γ_e. Formally, consider an edge $e = (v(\Box), v(\Box'))$ of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$. Suppose the crossings on γ_e are $c_1, \dots, c_r$, ordered from the $v(\Box)$ end to the $v(\Box')$ end. Then we include in P the edges $(v(\Box), c_1), (c_1, c_2), \dots, (c_{r-1}, c_r), (c_r, v(\Box'))$. After considering all edges of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$, we complete the construction of P. Note that η naturally induces a planar drawing of P (thus P is planar), which we denote by η_0. Clearly, the image of η_0 is equal to the image of η, which is $\varGamma$. See Figure 3 for an illustration of the construction of P. The following observation gives a relation between the treewidths of $G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}$ and P.

▶ **Observation 9.** $\mathbf{tw}(G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}) \leq O(\mathbf{tw}(P))$.

Based on the above observation, it now suffices to show that $\mathbf{tw}(P) = O(j)$. To this end, we need to introduce a notion called *vertex-face incidence graph*. We consider the plane-embedded graph (P, η_0). The *vertex-face incidence graph* P^+ of (P, η_0) is a bipartite graph defined as follows. One part of P^+ consists of the vertices of (P, η_0), while the other part consists of the faces of (P, η_0). A vertex v of (P, η_0) and a face F of (P, η_0) are connected by an edge in P^+ if v is incident to F. Let o be the outer face of (P, η_0), which is a vertex of P^+. The *depth* of a vertex v in (P, η_0) is defined as the shortest-path distance between o and v in P^+. It is well-known that $\mathbf{tw}(P)$ is linear in the maximum depth of a vertex in (P, η_0); see for example [5]. So we only need to show the depth of each vertex in (P, η_0) is $O(j)$.

Consider a vertex v of (P, η_0). By Observation 8, there exists a point b in the unbounded connected component of $\mathbb{R}^2 \backslash \varGamma$ and a path $f : [0, 1] \to \mathbb{R}^2$ from v to b such that $\Delta(f, \varGamma) = O(j)$. Suppose $\{x \in [0, 1] : f(x) \in \varGamma\} = \{x_1, \dots, x_k\}$ where $k = O(j)$ and $x_1 < \cdots < x_k$. We have $x_1 = 0$ because $f(0) = v \in \varGamma$. Let $I_i = \{x : x_i < x < x_{i+1}\}$ for $i \in [k-1]$ and $I_k = \{x : x_k < x \leq 1\}$. Since f is continuous, the image of each I_i under f is connected and disjoint from $\varGamma$, and hence lies in one face of (P, η_0), which we denote by F_i. We say two faces of (P, η_0) are *adjacent* if they are incident to a common vertex of (P, η_0). Clearly, the shortest-path distance between two adjacent faces of (P, η_0) in P^+ is 2. Note that for each $i \in [k-1]$, F_i and F_{i+1} are adjacent, as they are both incident to the point $f(x_{i+1}) \in \varGamma$, which is either a vertex of (P, η_0) or on an edge e of (P, η_0); in the latter case, F_i and F_{i+1}

are both incident to the two endpoints of e. Therefore, the shortest-path distance between F_1 and F_k in P^+ is at most $2k-2$, which is $O(j)$. Now F_1 is incident to $f(x_1) = f(0) = v$ and F_k is the outer face o of (P, η_0) since $b \in F_k$. It follows that the shortest-path distance between v and o is $O(j)$, and thus the depth of v is $O(j)$. This implies $\mathbf{tw}(P) = O(j)$ and hence $\mathbf{tw}(G_{\mathcal{L}_{\leq j}}/E^*_{\mathcal{L}_{\leq j}}) = O(j)$ by Observation 9. Property (iii) in Lemma 3 holds.

3.2 Constructing the partition $\{\mathcal{D}_1, \ldots, \mathcal{D}_p\}$

Given the layering ℓ of $\mathcal{D}$ presented in the previous section, we are able to construct the partition $\{\mathcal{D}_1, \ldots, \mathcal{D}_p\}$ of $\mathcal{D}$ in Theorem 2. The basic idea is similar to that used in Baker's technique: combining the congruent layers modulo p. Recall that $\mathcal{L}_1, \ldots, \mathcal{L}_m$ are the layers of $\mathcal{D}$. We define $\mathcal{D}_i = \bigcup_{j=0}^{\lfloor (m-i)/p \rfloor} \mathcal{L}_{jp+i}$, i.e., $\mathcal{D}_i$ consists of all layers whose index is congruent to i modulo p. Clearly, $\mathcal{D}_1, \ldots, \mathcal{D}_p$ can be computed in polynomial time. As $\{\mathcal{L}_1, \ldots, \mathcal{L}_m\}$ is a partition of $\mathcal{D}$, $\{\mathcal{D}_1, \ldots, \mathcal{D}_p\}$ is also a partition of $\mathcal{D}$. Also, since each $\mathcal{L}_i$ is a grid-respecting subset of $\mathcal{D}$, the partition $\{\mathcal{D}_1, \ldots, \mathcal{D}_p\}$ of $\mathcal{D}$ is grid-respecting. To prove Theorem 2, it suffices to show $\mathbf{tw}(G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i \backslash \mathcal{D}'})) = O(p + |\mathcal{D}'|)$ for any $i \in [p]$ and $\mathcal{D}' \subseteq \mathcal{D}_i$.

3.3 Bounding the treewidth when $\mathcal{D}' = \emptyset$

We first consider a special case of the treewidth bound in Theorem 2 where $\mathcal{D}' = \emptyset$. In other words, we prove $\mathbf{tw}(G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})) = O(p)$ for any $i \in [p]$. The argument for this is similar to the one used in [5] for planar graphs. So we only sketch the high-level ideas and the details will appear in the full paper. For simplicity, let us just consider the case $i = p$. Define $r = \lfloor m/p \rfloor + 1$ and $i_j = (j-1) \cdot p$ for $j \in \mathbb{N}$. So we have $\mathcal{D}_p = \bigcup_{j=1}^{r} \mathcal{L}_{i_j}$. We define a support tree T_{supp} as follows. The depth of T_{supp} is r. The root (i.e., the node at the 0-th level) of T_{supp} is a dummy node. For all $j \in [r]$, the nodes at the j-th level of T_{supp} are one-to-one corresponding to the connected components of $G_{\mathcal{L}_{>i_j}}$. The parent of the nodes at the first level is just the root. Consider a node $t \in T_{\mathsf{supp}}$ at the j-th level for $j \geq 2$. Since $G_{\mathcal{L}_{>i_j}}$ is a subgraph of $G_{\mathcal{L}_{>i_{j-1}}}$, the connected component of $G_{\mathcal{L}_{>i_j}}$ corresponding to t is contained in a unique connected component of $G_{\mathcal{L}_{>i_j}}$, which corresponds to a node t' at the $(j-1)$-th level of T_{supp}. We then define the parent of t as t'. For each node $t \in T_{\mathsf{supp}}$, we associate to t a set $\mathcal{A}_t \subseteq \mathcal{D}$ defined as follows. If t is the root, $\mathcal{A}_t = \emptyset$. Suppose t is at the j-th level for $j \in [r]$ and let $\mathcal{C}_t \subseteq \mathcal{L}_{>i_j}$ be the vertex set of the connected component of $G_{\mathcal{L}_{>i_j}}$ corresponding to t. Then we define $\mathcal{A}_t = \{D \in \mathcal{C}_t : i_j < \ell(D) \leq i_{j+1}\}$, i.e., $\mathcal{A}_t$ consists of all unit disks in $\mathcal{C}_t$ which lie in the layers $\mathcal{L}_{i_j+1}, \ldots, \mathcal{L}_{i_{j+1}}$. We then carefully use the three properties shown in Lemma 3 to argue that $\{\mathcal{A}_t\}_{t \in T_{\mathsf{supp}}}$ is a grid-respecting partition of $\mathcal{D}$, and $G_{\mathcal{A}_t}$ is adjacent to $G_{\mathcal{A}_{t'}}$ only if t and t' is adjacent in T. Property (iii) implies that each graph $J_t = G_{\mathcal{A}_t}/(E^*_{\mathcal{A}_t} \cup E_{\mathcal{A}_t \cap \mathcal{D}_p})$ has treewidth $O(p)$. Using this fact, we construct an $O(p)$-width tree decomposition for $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})$ by "gluing" $O(p)$-width tree decompositions for the graphs J_t along the edges of T_{supp}. This eventually implies $\mathbf{tw}(G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_p})) = O(p)$.

3.4 Handling the general case

In the previous section, we have proved that the partition $\{\mathcal{D}_1, \ldots, \mathcal{D}_p\}$ satisfies the condition in Theorem 2 for the special case where $\mathcal{D}' = \emptyset$. In this section, we shall consider the general case and complete the proof for Theorem 2. Let $i \in [p]$. Our goal is to show $\mathbf{tw}(G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i \backslash \mathcal{D}'})) = O(p + |\mathcal{D}'|)$ for every $\mathcal{D}' \subseteq \mathcal{D}_i$, knowing $\mathbf{tw}(G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})) = O(p)$.

For convenience, we denote by V the vertex set of $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})$ and V' the vertex set of $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i \backslash \mathcal{D}'})$. Since $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})$ is obtained from $G_{\mathcal{D}}$ via edge contraction, there is a corresponding quotient map $\pi : \mathcal{D} \to V$. Similarly, there is a quotient map

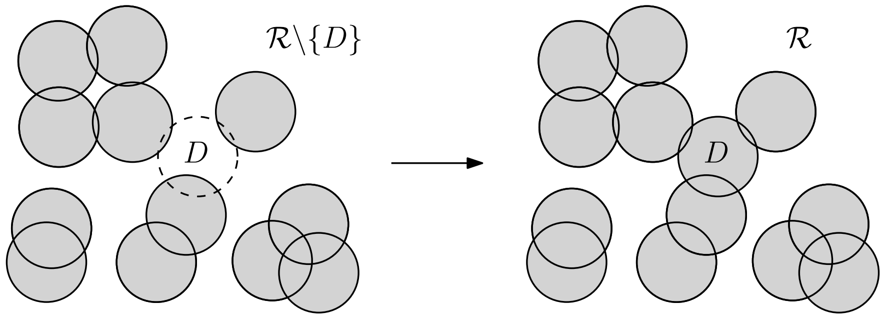

Figure 4 The three components of $G_{\mathcal{R}\backslash\{D\}}$ hit by D are merged into one connected component in $G_{\mathcal{R}}$, while the others remain the same.

$\pi' : \mathcal{D} \to V'$ corresponding to the edge contraction for obtaining $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i\backslash\mathcal{D}'})$. Note that $E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i\backslash\mathcal{D}'} \subseteq E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i}$. So there exists a unique map $\rho : V' \to V$ such that $\pi = \rho \circ \pi'$, and $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})$ can be viewed as a graph obtained from $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i\backslash\mathcal{D}'})$ via edge contraction with quotient map ρ.

As $\mathbf{tw}(G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})) = O(p)$, there exists a tree decomposition (T, β) of $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})$ of width $O(p)$. We define a map $\beta' : T \to 2^{V'}$ as $\beta'(t) = \rho^{-1}(\beta(t))$ for all nodes $t \in T$. By Fact 1, (T, β') is a tree decomposition of $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i\backslash\mathcal{D}'})$. Now it suffices to show that the width of this tree decomposition is $O(p + |\mathcal{D}'|)$. To this end, we establish a basic property of unit-disk graphs. For a graph G, we use the notation $\|G\|$ to denote the number of connected components of G. We have the following lemma.

▶ **Lemma 10.** *For a set $\mathcal{R}$ of unit disks and $\mathcal{R}' \subseteq \mathcal{R}$, $\|G_{\mathcal{R}\backslash\mathcal{R}'}\| - \|G_{\mathcal{R}}\| = O(|\mathcal{R}'|)$.*

Proof. We show that $\|G_{\mathcal{R}\backslash\{D\}}\| - \|G_{\mathcal{R}}\| = O(1)$ for any unit disk $D \in \mathcal{R}$. Then the lemma can be proved via a simple induction argument. We say D *hits* a connected component of $G_{\mathcal{R}\backslash\{D\}}$ if D intersects some unit disk in this connected component. Note that all connected components of $G_{\mathcal{R}\backslash\{D\}}$ hit by D are merged into one connected component in $G_{\mathcal{R}}$, and all the other connected components of $G_{\mathcal{R}\backslash\{D\}}$ remain the same in $G_{\mathcal{R}}$. See Figure 4 for an example. Thus, the quantity $\|G_{\mathcal{R}\backslash\{D\}}\| - \|G_{\mathcal{R}}\|$ is equal to the number of connected components of $G_{\mathcal{R}\backslash\{D\}}$ hit by D minus 1. So it suffices to show that D only hits $O(1)$ connected components of $G_{\mathcal{R}\backslash\{D\}}$. Suppose D hits k connected components of $G_{\mathcal{R}\backslash\{D\}}$. Pick a unit disk from each such connected component, and let $D_1, \dots, D_k$ be these unit disks. Note that $D_1, \dots, D_k$ are disjoint as they are from different connected components of $G_{\mathcal{R}\backslash\{D\}}$. On the other hand, $D_1, \dots, D_k$ are all contained in the disk D^+ centered at $\mathsf{ctr}(D)$ of radius 3, as they intersect D. The area of D^+ is 9π, so it can contain at most 9 disjoint unit disks. Thus, $k = O(1)$. ◀

Using the above lemma, we show that $|\rho^{-1}(U)| = O(|U| + |\mathcal{D}'|)$ for any $U \subseteq V$. Since $\mathcal{D}_i$ is a grid-respecting subset of $\mathcal{D}$, for each $v \in V$, $\pi^{-1}(\{v\})$ is either (the vertex set of) a cell clique of $G_{\mathcal{D}}$ that is disjoint from $\mathcal{D}_i$ or (the vertex set of) a connected component of $G_{\mathcal{D}_i}$; we say v is a *type-1* vertex in the former case and a *type-2* vertex in the latter case. Let U_1 (resp., U_2) be the type-1 (resp., type-2) vertices in U. For each $u \in U_1$, we have $|\rho^{-1}(\{u\})| = |\pi'(\pi^{-1}(\{u\}))| = 1$, as every cell clique of $G_{\mathcal{D}}$ is contracted into one vertex in $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i\backslash\mathcal{D}'})$. Thus, $|\rho^{-1}(U_1)| = |U_1|$. To bound $|\rho^{-1}(U_2)|$, we consider $\pi^{-1}(U_2) \subseteq \mathcal{D}$. By definition, $\pi^{-1}(\{u\})$ is a connected component of $G_{\mathcal{D}_i}$ for each $u \in U_2$, and thus $\|G_{\pi^{-1}(U_2)}\| = |U_2|$. Set $\mathcal{I} = \pi^{-1}(U_2) \cap \mathcal{D}'$. By Lemma 10, we have

$$\|G_{\pi^{-1}(U_2)\backslash\mathcal{D}'}\| - \|G_{\pi^{-1}(U_2)}\| = \|G_{\pi^{-1}(U_2)\backslash\mathcal{I}}\| - \|G_{\pi^{-1}(U_2)}\| = O(|\mathcal{I}|),$$

which implies $\|G_{\pi^{-1}(U_2)\backslash\mathcal{D}'}\| = O(|U_2|+|\mathcal{D}'|)$ because $|\mathcal{I}| \le |\mathcal{D}'|$. Since $\pi^{-1}(U_2)\backslash\mathcal{D}' \subseteq \mathcal{D}_i\backslash\mathcal{D}'$, π' maps the vertices in each connected component of $G_{\pi^{-1}(U_2)\backslash\mathcal{D}'}$ to the same vertex in V'. Therefore, $|\pi'(\pi^{-1}(U_2)\backslash\mathcal{D}')| \le \|G_{\pi^{-1}(U_2)\backslash\mathcal{D}'}\| = O(|U_2|+|\mathcal{D}'|)$. Now we have the inequality

$$|\pi'(\pi^{-1}(U_2))| \le |\pi'(\pi^{-1}(U_2)\backslash\mathcal{D}')| + |\pi'(\mathcal{D}')| = O(|U_2|+|\mathcal{D}'|).$$

It follows that $|\rho^{-1}(U_2)| = O(|U_2|+|\mathcal{D}'|)$, and thus $|\rho^{-1}(U)| = O(|U|+|\mathcal{D}'|)$. As a result, for all $t \in T$, $|\beta'(t)| = |\rho^{-1}(\beta(t))| = O(|\beta(t)|+|\mathcal{D}'|) = O(p+|D'|)$. So (T, β') is a tree decomposition of $G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i\backslash\mathcal{D}'})$ of width $O(p+|D'|)$, completing the proof of Theorem 2.

4 Applications

4.1 Contraction decomposition for unit-disk graphs

In this section, we use Theorem 2 to prove the first Contraction Decomposition Theorem (CDT) for unit-disk graphs, which is shown below.

▶ **Theorem 11** (Contraction Decomposition Theorem). *Given a set $\mathcal{D}$ of n unit disks and an integer $p \in [n]$, one can compute in polynomial time a partition $\{E_1, \dots, E_p\}$ of $E_{\mathcal{D}}$ such that for every $i \in [p]$, $\mathbf{tw}(G_{\mathcal{D}}/E_i) = O(p^2)$.*

To prove the above theorem, it suffices to compute in polynomial time p *disjoint* subsets $E_1, \dots, E_p \subseteq E_{\mathcal{D}}$ such that $\mathbf{tw}(G_{\mathcal{D}}/E_i) = O(p^2)$ for every $i \in [p]$ (that is, we do not require $\{E_1, \dots, E_p\}$ to be a partition of $E_{\mathcal{D}}$), as contracting more edges only decreases the treewidth.

We start by applying the algorithm of Theorem 2 on $\mathcal{D}$ to obtain in polynomial time a grid-respecting partition $\{\mathcal{D}_1, \dots, \mathcal{D}_p\}$ of $\mathcal{D}$. Consider any $i \in [p]$. Setting $\mathcal{D}' = \emptyset$ in Theorem 2 gives us $\mathbf{tw}(G_{\mathcal{D}}/(E^*_{\mathcal{D}} \cup E_{\mathcal{D}_i})) = O(p)$. We are going to use this fact later in our analysis. Next, we state a lemma which will be used in our construction of the edge sets $E_1, \dots, E_p$.

▶ **Lemma 12.** *The edge set of a clique K of size larger than $4p$ can be partitioned in polynomial time into p parts such that each part contains a spanning tree of K.*

We construct the edge sets $E_1, \dots, E_p$ in the following way. Consider any edge $e = (u, v) \in E_{\mathcal{D}}$. If $u \in \mathcal{D}_i$ and $v \in \mathcal{D}_j$ for $i \neq j$, then we totally ignore e (i.e., do not add it to any of $E_1, \dots, E_p$). Otherwise, let $u, v \in \mathcal{D}_i$ for some $i \in [p]$. If e is not a part of any cell clique, we add e to the part E_i. If e is a part of a cell clique of size at most $4p$, we also add e to the part E_i. The only remaining edges are those in the cell cliques of size larger than $4p$. Consider any such cell clique K. Using the algorithm in Lemma 12, we partition the edge set of K into exactly p parts $H_1, \dots, H_p$ each of which contains a spanning tree of K, and then add the edges in H_i to E_i for $i \in [p]$. This completes the construction of $E_1, \dots, E_p \subseteq E_{\mathcal{D}}$. It is clear that $E_1, \dots, E_p$ are disjoint. Now it suffices to bound $\mathbf{tw}(G_{\mathcal{D}}/E_i)$ for every $i \in [p]$.

▶ **Lemma 13.** *For all $i \in [p]$, $\mathbf{tw}(G_{\mathcal{D}}/E_i) = O(p^2)$.*

4.2 Near-optimal bipartization for unit-disk graphs

In this section, we use Theorem 2 to solve Bipartization on unit-disk graphs. Due to limited space, we only provide a high-level description of our algorithm with details omitted. Let $\mathcal{D}$ be a set of n unit disks and k be the parameter. Recall that we want to find $\mathcal{X} \subseteq \mathcal{D}$ of size at most k such that $G_{\mathcal{D}\backslash\mathcal{X}}$ is bipartite. We refer to such a set $\mathcal{X}$ as an *OCT*.

An easy but crucial remark is that, for every clique K in $G_{\mathcal{D}}$, an OCT contains all vertices of K except at most two. We start by checking if there is some cell clique in $G_{\mathcal{D}}$ with size at least $k+3$, in which case it trivially answers NO. From now on, we may assume all cell cliques have size at most $k+2$. The first step of our algorithm is to apply the following randomized algorithm to obtain a small candidate set $\mathsf{Cand} \subseteq \mathcal{D}$ for OCT. This can be done via the technique of representative sets, see Lemma 5 in [5] for more details.

▶ **Lemma 14.** *Given a graph $G=(V,E)$ and a number k, one can compute $\mathsf{Cand} \subseteq V$ of size $k^{O(1)}$ such that G has an OCT of size k iff G has an OCT of size k that is a subset of Cand, using a polynomial-time randomized algorithm with success probability $1-1/2^{|V|}$.*

By the above lemma, $|\mathsf{Cand}| = k^{O(1)}$ and it suffices to find an OCT $\mathcal{X} \subseteq \mathsf{Cand}$ of $G_\mathcal{D}$ of size at most k. Suppose such an OCT $\mathcal{X}$ exists (but is unknown to us). Next, we apply the algorithm of Theorem 2 with $p = \lfloor \sqrt{k} \rfloor$ to obtain the grid-respecting partition $\{\mathcal{D}_1, \dots, \mathcal{D}_p\}$ of $\mathcal{D}$ in polynomial time. As $|\mathcal{X}| \leq k$ and $\{\mathcal{D}_1, \dots, \mathcal{D}_p\}$ is a partition of $\mathcal{D}$, there exists an index $i \in [p]$ such that $|\mathcal{D}_i \cap \mathcal{X}| \leq k/p$. By trying all indices in $[p]$, we can assume that the algorithm knows the index i. Moreover, we know that $\mathcal{D}_i \cap \mathcal{X} \subseteq \mathcal{D}_i \cap \mathsf{Cand}$ as $\mathcal{X} \subseteq \mathsf{Cand}$. Thus, by trying all the subsets of $\mathcal{D}_i \cap \mathsf{Cand}$ of size at most k/p, we can assume that the algorithm knows $\mathcal{S} = \mathcal{D}_i \cap \mathcal{X}$; note that the number of such subsets is $|\mathsf{Cand}|^{O(k/p)} = 2^{O(\sqrt{k}\log k)}$. The above is a variant of Baker's technique, which is also used in [5].

Now it suffices to find an OCT $\mathcal{X}$ of size at most k which contains $\mathcal{S}$ but is disjoint from $\mathcal{D}_i \backslash \mathcal{S}$. By Theorem 2, we have $\mathbf{tw}(G_\mathcal{D}/(E^*_\mathcal{D} \cup E_{\mathcal{D}_i \backslash \mathcal{S}})) = O(p + |\mathcal{S}|) = O(\sqrt{k})$. Let (T, β^*) be a tree decomposition of $G_\mathcal{D}/(E^*_\mathcal{D} \cup E_{\mathcal{D}_i \backslash \mathcal{S}})$ of width $O(\sqrt{k})$. We can then use Fact 1 to obtain a tree decomposition (T, β) of $G_\mathcal{D}$ from (T, β^*). Then we compute the OCT $\mathcal{X}$ via dynamic programming on (T, β). The main difficulty here is that although the width of (T, β^*) is $O(\sqrt{k})$, the width of (T, β) is unbounded. Fortunately, we can exploit the $O(\sqrt{k})$ width of (T, β^*) to show that the size of the DP table at each node $t \in T$ as well as the total number of different DP configurations to be considered are both bounded by $2^{O(\sqrt{k}\log k)}$. The main reason is that (essentially) each vertex of $G_\mathcal{D}/(E^*_\mathcal{D} \cup E_{\mathcal{D}_i \backslash \mathcal{S}})$ corresponds to either a cell clique in $G_\mathcal{D}$ or a connected component of $G_{\mathcal{D}_i \backslash \mathcal{S}}$. A cell clique K can have $O(k^2)$ different possible configurations in the solution as by assumption the size of K is $O(k)$ and at most two vertices in K are not in the OCT. A connected component of $G_{\mathcal{D}_i \backslash \mathcal{S}}$ can only have two different configurations as nothing in $\mathcal{D}_i \backslash \mathcal{S}$ is contained in the OCT and a connected graph can have at most two different 2-colorings. As such, we can do DP on (T, β) in $2^{O(\sqrt{k}\log k)} n^{O(1)}$ time despite of its unbounded width. The details of our algorithm will appear in the full paper. Also, the generalization of our algorithm to Group Feedback Vertex Set is deferred to the full version.

▶ **Theorem 15.** *There exists a randomized algorithm that solves, for given a set $\mathcal{D}$ of n unit disks in the plane and a number k, the* Bipartization *problem on $G_\mathcal{D}$ in $2^{O(\sqrt{k}\log k)} n^{O(1)}$ time, with success probability at least $1 - 1/2^{|\mathcal{D}|}$.*

We show that the algorithm in the above theorem is near optimal. Specifically, we cannot hope for a $2^{o(\sqrt{k})} n^{O(1)}$ running time, assuming ETH.

▶ **Theorem 16.** *Assuming the ETH,* Bipartization *on unit-disk graphs cannot be solved in $2^{o(\sqrt{k})} n^{O(1)}$ time, where k is the solution size and n is the number of vertices.*

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 *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 573–581, 2005.

2 Jochen Alber, Henning Fernau, and Rolf Niedermeier. Graph separators: a parameterized view. *J. Comput. Syst. Sci.*, 67(4):808–832, 2003. doi:10.1016/S0022-0000(03)00072-2.

3 Jochen Alber and Jirí Fiala. Geometric separation and exact solutions for the parameterized independent set problem on disk graphs. *J. Algorithms*, 52(2):134–151, 2004. doi:10.1016/j.jalgor.2003.10.001.

4 Shinwoo An and Eunjin Oh. Feedback vertex set on geometric intersection graphs. In Hee-Kap Ahn and Kunihiko Sadakane, editors, *32nd International Symposium on Algorithms and Computation, ISAAC 2021, December 6-8, 2021, Fukuoka, Japan*, volume 212 of *LIPIcs*, pages 47:1–47:12. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ISAAC.2021.47.

5 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. *CoRR, to appear in SODA 2022*, abs/2111.14196, 2021. arXiv:2111.14196.

6 Thang Nguyen Bui and Andrew Peck. Partitioning planar graphs. *SIAM J. Comput.*, 21(2):203–215, 1992. doi:10.1137/0221016.

7 Sergio Cabello and Miha Jejčič. Shortest paths in intersection graphs of unit disks. *Computational Geometry*, 48(4):360–367, 2015.

8 Timothy M Chan and Dimitrios Skrepetos. All-pairs shortest paths in geometric intersection graphs. In *Workshop on Algorithms and Data Structures*, pages 253–264. Springer, 2017.

9 Timothy M Chan and Dimitrios Skrepetos. Approximate shortest paths and distance oracles in weighted unit-disk graphs. *Journal of Computational Geometry (Old Web Site)*, 10(2):3–20, 2019.

10 Brent N Clark, Charles J Colbourn, and David S Johnson. Unit disk graphs. *Discrete mathematics*, 86(1-3):165–177, 1990.

11 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

12 Mark de Berg, Hans L Bodlaender, Sándor Kisfaludi-Bak, Dániel Marx, and Tom C van der Zanden. A framework for eth-tight algorithms and lower bounds in geometric intersection graphs. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 574–586, 2018.

13 Erik D. Demaine, Fedor V. Fomin, Mohammad Taghi Hajiaghayi, and Dimitrios M. Thilikos. Subexponential parameterized algorithms on bounded-genus graphs and H-minor-free graphs. *J. ACM*, 52(6):866–893, 2005. doi:10.1145/1101821.1101823.

14 Erik D. Demaine, MohammadTaghi Hajiaghayi, and Ken-ichi Kawarabayashi. Contraction decomposition in h-minor-free graphs and algorithmic applications. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pages 441–450, 2011.

15 Erik D. Demaine, MohammadTaghi Hajiaghayi, and Bojan Mohar. Approximation algorithms via contraction decomposition. *Combinatorica*, 30(5):533–552, 2010.

16 Frederic Dorn, Fedor V. Fomin, Daniel Lokshtanov, Venkatesh Raman, and Saket Saurabh. Beyond bidimensionality: Parameterized subexponential algorithms on directed graphs. *Inf. Comput.*, 233:60–70, 2013. doi:10.1016/j.ic.2013.11.006.

17 Samuel Fiorini, Nadia Hardy, Bruce Reed, and Adrian Vetta. Planar graph bipartization in linear time. *Discrete Applied Mathematics*, 156(7):1175–1180, 2008.

18 Fedor V. Fomin, Daniel Lokshtanov, Sudeshna Kolay, Fahad Panolan, and Saket Saurabh. Subexponential algorithms for rectilinear steiner tree and arborescence problems. *ACM Trans. Algorithms*, 16(2):21:1–21:37, 2020. doi:10.1145/3381420.

19 Fedor V. Fomin, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. Subexponential parameterized algorithms for planar and apex-minor-free graphs via low treewidth pattern covering. In Irit Dinur, editor, *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 515–524. IEEE Computer Society, 2016. doi:10.1109/FOCS.2016.62.

20 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Decomposition of map graphs with applications. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, volume 132 of *LIPIcs*, pages 60:1–60:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.ICALP.2019.60.

21 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.

22 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Eth-tight algorithms for long path and cycle on unit disk graphs. In Sergio Cabello and Danny Z. Chen, editors, *36th International Symposium on Computational Geometry, SoCG 2020, June 23-26, 2020, Zürich, Switzerland*, volume 164 of *LIPIcs*, pages 44:1–44:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.SoCG.2020.44.

23 Fedor V. Fomin, Daniel Lokshtanov, and Saket Saurabh. Excluded grid minors and efficient polynomial-time approximation schemes. *J. ACM*, 65(2):10:1–10:44, 2018. doi:10.1145/3154833.

24 Jie Gao and Li Zhang. Well-separated pair decomposition for the unit-disk graph metric and its applications. *SIAM Journal on Computing*, 35(1):151–169, 2005.

25 Michel X Goemans and David P Williamson. Primal-dual approximation algorithms for feedback problems in planar graphs. *Combinatorica*, 18(1):37–59, 1998.

26 MohammadTaghi Hajiaghayi. Contraction and minor graph decomposition and their algorithmic applications. *Filmed Talk at Microsoft Research*, 2016.

27 William K Hale. Frequency assignment: Theory and applications. *Proceedings of the IEEE*, 68(12):1497–1514, 1980.

28 Falk Hüffner. Algorithm engineering for optimal graph bipartization. *Journal of Graph Algorithms and Applications*, 13(2):77–98, 2009.

29 Bart M. P. Jansen, Marcin Pilipczuk, and Erik Jan van Leeuwen. A deterministic polynomial kernel for odd cycle transversal and vertex multiway cut in planar graphs. In Rolf Niedermeier and Christophe Paul, editors, *36th International Symposium on Theoretical Aspects of Computer Science, STACS 2019, March 13-16, 2019, Berlin, Germany*, volume 126 of *LIPIcs*, pages 39:1–39:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.STACS.2019.39.

30 Ken-ichi Kawarabayashi and Bruce Reed. An (almost) linear time algorithm for odd cycles transversal. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 365–378. SIAM, 2010.

31 Philip N. Klein. A subset spanner for planar graphs, : with application to subset TSP. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*, pages 749–756, 2006. doi:10.1145/1132516.1132620.

32 Philip N. Klein. A linear-time approximation scheme for TSP in undirected planar graphs with edge-weights. *SIAM J. Comput.*, 37(6):1926–1952, 2008. doi:10.1137/060649562.

33 Philip N. Klein and Dániel Marx. Solving planar k -terminal cut in $O(n^{c\sqrt{k}})$ time. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part I*, volume 7391 of *Lecture Notes in Computer Science*, pages 569–580. Springer, 2012. doi:10.1007/978-3-642-31594-7_48.

34 Philip N. Klein and Dániel Marx. A subexponential parameterized algorithm for subset TSP on planar graphs. In Chandra Chekuri, editor, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 1812–1830. SIAM, 2014. doi:10.1137/1.9781611973402.131.

35 Stefan Kratsch and Magnus Wahlström. Compression via matroids: A randomized polynomial kernel for odd cycle transversal. *ACM Trans. Algorithms*, 10(4):20:1–20:15, 2014. doi:10.1145/2635810.

36 Daniel Lokshtanov, NS Narayanaswamy, Venkatesh Raman, MS Ramanujan, and Saket Saurabh. Faster parameterized algorithms using linear programming. *ACM Transactions on Algorithms (TALG)*, 11(2):1–31, 2014.

37 Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Jie Xue, and Meirav Zehavi. Subexponential parameterized algorithms on disk graphs. *to appear in SODA 2022*, 2021.

38 Daniel Lokshtanov, Saket Saurabh, and Magnus Wahlström. Subexponential parameterized odd cycle transversal on planar graphs. In Deepak D'Souza, Telikepalli Kavitha, and Jaikumar Radhakrishnan, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2012, December 15-17, 2012, Hyderabad, India*, volume 18 of *LIPIcs*, pages 424–434. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2012. doi:10.4230/LIPIcs.FSTTCS.2012.424.

39 Dániel Marx, Pranabendu Misra, Daniel Neuen, and Prafullkumar Tale. A framework for parameterized subexponential algorithms for generalized cycle hitting problems on planar graphs. *CoRR, to appear in SODA 2022*, abs/2110.15098, 2021. arXiv:2110.15098.

40 Dániel Marx, Marcin Pilipczuk, and Michal Pilipczuk. On subexponential parameterized algorithms for steiner tree and directed subset TSP on planar graphs. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 474–484. IEEE Computer Society, 2018. doi:10.1109/FOCS.2018.00052.

41 Dániel Marx and Michal Pilipczuk. Optimal parameterized algorithms for planar facility location problems using voronoi diagrams. In Nikhil Bansal and Irene Finocchi, editors, *Algorithms - ESA 2015 - 23rd Annual European Symposium, Patras, Greece, September 14-16, 2015, Proceedings*, volume 9294 of *Lecture Notes in Computer Science*, pages 865–877. Springer, 2015. doi:10.1007/978-3-662-48350-3_72.

42 Jesper Nederlof. Detecting and counting small patterns in planar graphs in subexponential parameterized time. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 1293–1306. ACM, 2020. doi:10.1145/3357713.3384261.

43 Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Contraction decomposition in unit disk graphs and algorithmic applications in parameterized complexity. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1035–1054. SIAM, 2019. doi:10.1137/1.9781611975482.64.

44 Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Parameterized computational geometry via decomposition theorems. In Gautam K. Das, Partha Sarathi Mandal, Krishnendu Mukhopadhyaya, and Shin-Ichi Nakano, editors, *WALCOM: Algorithms and Computation - 13th International Conference, WALCOM 2019, Guwahati, India, February 27 - March 2, 2019, Proceedings*, volume 11355 of *Lecture Notes in Computer Science*, pages 15–27. Springer, 2019. doi:10.1007/978-3-030-10564-8_2.

45 Bruce Reed, Kaleigh Smith, and Adrian Vetta. Finding odd cycle transversals. *Operations Research Letters*, 32(4):299–301, 2004.

46 Siamak Tazari. Faster approximation schemes and parameterized algorithms on (odd-)h-minor-free graphs. *Theor. Comput. Sci.*, 417:95–107, 2012. doi:10.1016/j.tcs.2011.09.014.

47 Haitao Wang and Jie Xue. Near-optimal algorithms for shortest paths in weighted unit-disk graphs. *Discrete & Computational Geometry*, 64(4):1141–1166, 2020.

48 Mihalis Yannakakis. Node-and edge-deletion np-complete problems. In *Proceedings of the tenth annual ACM symposium on Theory of computing*, pages 253–264, 1978.

49 Yu-Shuan Yeh, Joanne C Wilson, and Stuart C Schwartz. Outage probability in mobile telephony with directive antennas and macrodiversity. *IEEE transactions on vehicular technology*, 33(3):123–127, 1984.