Maximum Independent Sets in Disk Graphs with Disks in Convex Position

Anastasiia Tkachenko
Kahlert School of Computing, University of Utah, Salt Lake City, UT, USA

Haitao Wang
Kahlert School of Computing, University of Utah, Salt Lake City, UT, USA

Abstract

For a set $\mathcal{D}$ of disks in the plane, its disk graph $G(\mathcal{D})$ is the graph with vertex set $\mathcal{D}$, where two vertices are adjacent if and only if the corresponding disks intersect. Given a set $\mathcal{D}$ of n weighted disks, computing a maximum independent set of $G(\mathcal{D})$ is NP-hard. In this paper, we present an $O(n^3 \log n)$-time algorithm for this problem in a special setting in which the disks are in *convex position*, meaning that every disk appears on the convex hull of $\mathcal{D}$. This setting has been studied previously for disks of equal radius, for which an $O(n^{37/11})$-time algorithm was known. Our algorithm also works in the weighted case where disks have weights and the goal is to compute a maximum-weight independent set. As an application of our result, we obtain an $O(n^3 \log^2 n)$-time algorithm for the *dispersion problem* on a set of n disks in convex position: given an integer k, compute a subset of k disks that maximizes the minimum pairwise distance among all disks in the subset.

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

Keywords and phrases disk graphs, independent sets, convex position, dispersion

Digital Object Identifier 10.4230/LIPIcs.SWAT.2026.40

Related Version *Full Version*: `https://arxiv.org/abs/2604.10828`

1 Introduction

An *independent set* in a graph is a subset of vertices with no edges between any two of them. The *maximum independent set problem* asks for an independent set of maximum cardinality. In weighted graphs, where each vertex is assigned a weight, the *maximum-weight independent set problem* seeks an independent set of maximum total weight.

In general graphs, these problems are computationally intractable [20]. This hardness has motivated research along two complementary directions. One direction focuses on designing approximation algorithms, which trade exactness for efficiency. The other aims to identify restricted settings, such as specific graph classes or additional structural constraints, under which exact polynomial-time algorithms become possible. Exact polynomial-time algorithms are known for a variety of restricted graph classes, including circular [5, 17], chordal [16], trapezoid [15], outerstring [7], and Burling graphs [28], where the additional structure can be exploited algorithmically.

In light of the above, geometric intersection graphs are of particular interest [1]. In these graphs, vertices represent geometric objects and edges encode their intersections. In this work, we focus on *disk graphs*, the intersection graphs of disks, which are among the most extensively studied geometric graph families. Beyond their intrinsic theoretical interest, disk graphs provide a natural abstraction for wireless and sensor networks, where connectivity is governed by transmission ranges that are commonly modeled as disks [6, 12, 24, 25].

20th Scandinavian Symposium on Algorithm Theory (SWAT 2026).
Editor: Pierre Fraigniaud; Article No. 40; pp. 40:1–40:18
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

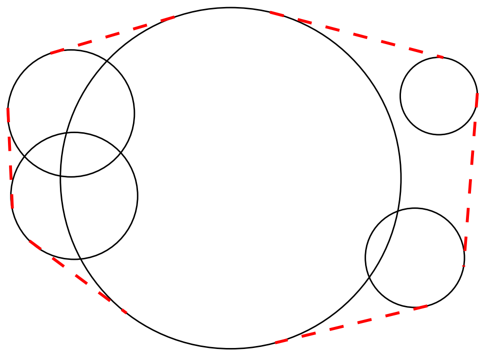

Figure 1 Illustrating a set of disks in convex position. The red dashed segments are on the boundary of the convex hull of all disks.

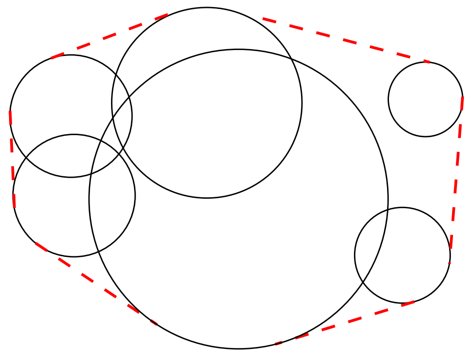

Figure 2 Illustrating a set of disks in strongly convex position. The red dashed segments are on the boundary of the convex hull of all disks.

Formally, let $\mathcal{D}$ be a set of disks in the plane. The *disk graph* $G(\mathcal{D})$ is the graph with vertex set $\mathcal{D}$, where two vertices are adjacent if and only if the corresponding disks intersect. Thus, a subset $\mathcal{D}' \subseteq \mathcal{D}$ is an independent set in $G(\mathcal{D})$ if and only if the disks in $\mathcal{D}'$ are pairwise disjoint. In addition, each disk is assigned a weight.

The maximum independent set problem in disk graphs remains NP-hard even when all disks have the same radius, in which case the graph is a *unit-disk graph* [12]. Many approximation algorithms for this problem have been developed; see, for example, [1, 9, 13, 14, 22, 23]. Beyond their algorithmic interest, independent sets in intersection graphs arise in several applications, such as map labeling in computational cartography [3]. For instance, in map labeling one is given a collection of candidate label regions and seeks a largest subset that can be placed without overlap; selecting such a subset is precisely a maximum independent set problem in the corresponding intersection graph.

These results naturally raise the question of whether, and to what extent, geometric structure can impose sufficient constraints to render otherwise intractable combinatorial problems tractable.

1.1 Our result

In this paper, we consider a special setting in which the disks in $\mathcal{D}$ are in *convex position*, meaning that every disk of $\mathcal{D}$ contributes to the boundary of the convex hull of $\mathcal{D}$ (see Figure 1). Under this assumption, we present an exact algorithm that computes a maximum-weight independent set in $G(\mathcal{D})$ in $O(n^3 \log n)$ time. This setting bridges the gap between the general two-dimensional problem and one-dimensional variants such as circular graphs, which admit near-linear-time algorithms [5].

The maximum-weight independent set problem under this convex position assumption has been studied previously, but only for the unit-disk case in which all disks have the same radius [30, 33]; in that setting, the previously best algorithm runs in $O(n^{37/11})$ time [33]. Our algorithm applies to disks of arbitrary radii and, moreover, improves the running time even in the unit-disk case.

We say that a set of disks is in *strongly convex position* if every disk has a single maximal disk arc appearing on the boundary of the convex hull of all disks (see Figure 2; but the disks in Figure 1 are not in strongly convex position). We solve the problem gradually as follows.

1. We first consider a very special case in which the disks of $\mathcal{D}$ are in strongly convex position and $\mathcal{D}$ has a maximum-weight independent set that is itself strongly convex; we refer to this as the *strongly-convex-input-and-output* case. Our algorithm for this case is based on dynamic programming and can be viewed as an extension of the unit-disk algorithm of [33] to disks of arbitrary radii. Moreover, we identify a new monotonicity property that allows our algorithm to run faster than the previous unit-disk algorithm [33]. Our algorithm for this case is presented in Section 3.
2. We then handle the *strongly-convex-input* case, in which $\mathcal{D}$ is strongly convex but does not necessarily admit a strongly convex maximum-weight independent set. We reduce this case to the first one by adding a set B of $O(n)$ auxiliary points, treated as special disks, such that (i) these special disks appear in any maximum-weight independent set of $\mathcal{D} \cup B$, and (ii) for any subset $\mathcal{D}' \subseteq \mathcal{D}$, the set $\mathcal{D}' \cup B$ is strongly convex. This case is discussed in Section 4.
3. Finally, we address the most general case in Section 5, where $\mathcal{D}$ is not necessarily strongly convex and does not necessarily admit a strongly convex maximum-weight independent set. We reduce this case to the second one by again adding a set of auxiliary points as special disks.

As an application of our independent set algorithm, we also solve a *dispersion problem* for a set of n disks in convex position: given an integer k, the goal is to find a subset of k disks that maximizes the minimum pairwise distance among all disks in the subset, where the distance between two disks is defined as the minimum distance between any point in one disk and any point in the other. Our algorithm for this problem runs in $O(n^3 \log^2 n)$ time and is presented in Section 6.

Due to the space limit, many proofs are omitted but can be found in the full paper.

1.2 Related work

Recent developments in geometric algorithms have demonstrated that the convex position constraint can serve as a powerful structural assumption. For a variety of classical problems, it enables exact polynomial-time algorithms, often by exploiting the inherent cyclic structure induced by convexity.

For example, the k-center problem is NP-hard for arbitrary point sets, but becomes polynomial-time solvable when the points are in convex position [11, 33]. Another recent result that exploits convex position concerns dominating sets in disk graphs [31, 33]. Note that for unit-disk graphs, the disk set is in convex position if and only if the centers of the disks are in convex position; this equivalence no longer holds when disks have different radii.

In the literature, related convexity assumptions have also been studied for disk graphs, including settings in which the disk centers are in convex position and settings in which the disks themselves are in convex position. For instance, Herrera and Pérez-Lantero [18] and Huemer and Pérez-Lantero [19] investigate combinatorial properties arising when the centers follow a prescribed convex pattern: each disk uses a distinct side of a convex polygon as its diameter, and its center is the midpoint of that side. Under the convex position assumption on the disks, it has also been reported that a maximum clique in the corresponding disk graph can be computed in polynomial time [8].

The convex position assumption has also been explored for classical problems that are already polynomial-time solvable in general, as it can simplify the structure and lead to faster algorithms. A well-known example is the linear-time algorithm for computing the Voronoi diagram of points in convex position [4]. Additional results for points in convex position can be found in [10, 18, 21, 27, 32].

A closely related problem is *dispersion* (also known as the *maximally separated set* problem [2]). Given a set of points in the plane and an integer k, the goal is to select k points that maximize their minimum pairwise distance; the problem is NP-hard in general [34]. When the points are in convex position, however, the problem becomes polynomial-time solvable [30, 33], and the previously best algorithm runs in $O(n^{37/11} \log n)$ time [33]. Our dispersion problem generalizes this setting from points to disks and our algorithm is even faster.

2 Preliminaries

In this section, we introduce notation and basic concepts used throughout the paper.

Let $\mathcal{D}$ denote a set of n disks D_i in the plane $\mathbb{R}^2$, for $1 \leq i \leq n$. For each disk D_i, let p_i be its center and let r_i be its radius. For any subset $\mathcal{D}' \subseteq \mathcal{D}$, we use $\mathcal{H}(\mathcal{D}')$ to denote the convex hull of $\mathcal{D}'$.

For any region R in the plane, let ∂R denote its boundary (e.g., $\partial\mathcal{H}(\mathcal{D})$ is the boundary of $\mathcal{H}(\mathcal{D})$ and ∂D is the boundary of a disk D). We say that the disks of $\mathcal{D}$ are in *convex position* if the boundary of every disk appears on $\partial\mathcal{H}(\mathcal{D})$, and they are in *strongly convex position* if the boundary of every disk has a single (maximal) arc appearing on $\partial\mathcal{H}(\mathcal{D})$. For simplicity, for any subset $\mathcal{D}' \subseteq \mathcal{D}$, if the disks of $\mathcal{D}'$ are in (strongly) convex position, we also say that $\mathcal{D}'$ is (strongly) convex.

For any two points p and q in the plane, let $\overline{pq}$ denote the line segment connecting them, and let $|pq|$ denote the Euclidean distance between p and q. For a point p with a *weight* $w(p)$, define

$$d_p(q) = |pq| + w(p)$$

for any point q in the plane. We call $d_p(q)$ the *weighted distance* between p and q.

For each disk $D_i \in \mathcal{D}$, we define the weight of its center p_i to be $-r_i$, i.e., the negative of the radius of D_i. Hence, for any point $q \in \mathbb{R}^2$, we have $d_{p_i}(q) = |p_i q| - r_i$. If q lies outside D_i, then $d_{p_i}(q)$ equals the minimum Euclidean distance between q and D_i. For this reason, we also refer to $d_{p_i}(q)$ as the *distance* between q and D_i (noting that this distance is negative when q lies in the interior of D_i).

Two disks $D_i, D_j \in \mathcal{D}$ are said to be *disjoint* if $|p_i p_j| > r_i + r_j$. A subset $\mathcal{D}' \subseteq \mathcal{D}$ is an *independent set* if the disks in $\mathcal{D}'$ are pairwise disjoint.

Throughout the paper, we assume that the disks of $\mathcal{D}$ are in convex position (but not necessarily in strongly convex position), and that each disk D_i is assigned a weight W_i. Our goal is to compute a maximum-weight independent set of $\mathcal{D}$, that is, an independent set of disks whose total weight is maximized. Since disks with non-positive weight can be ignored, we assume $W_i > 0$ for all D_i. For any subset $\mathcal{D}' \subseteq \mathcal{D}$, we use $W(\mathcal{D}')$ to denote the sum of the weights of the disks in $\mathcal{D}'$.

For any disk D and any two points $a, b \in \partial D$, we use $\partial_{[a,b]} D$ to denote the portion of ∂D from a to b in counterclockwise order. Similarly, for any subset $\mathcal{D}' \subseteq \mathcal{D}$ and any two points $a, b \in \partial\mathcal{H}(\mathcal{D}')$, we use $\partial_{[a,b]}\mathcal{H}(\mathcal{D}')$ to denote the portion of $\partial\mathcal{H}(\mathcal{D}')$ from a to b in counterclockwise order.

Finally, for ease of exposition, we assume general position: no point in the plane is equidistant from four disks of $\mathcal{D}$, and no line is tangent to three disks.

3 The strongly-convex-input-and-output case

We consider the *strongly-convex-input-and-output* case, in which $\mathcal{D}$ is strongly convex and admits a maximum-weight independent set that is itself strongly convex. To simplify the discussion, we assume that each disk in $\mathcal{D}$ has positive radius (i.e., no disk degenerates to a point). The same algorithm applies when some disks are points, but handling such degeneracies would require more technical discussion. We present an algorithm that computes a maximum-weight independent set in $O(n^3 \log n)$ time.

We begin with the following lemma (see the full paper for the proof).

▶ **Lemma 1.** *For any subset of three disks $\{D_i, D_j, D_k\} \subseteq \mathcal{D}$ that forms a strongly convex independent set, there exists a unique point $v_{ijk} \in \mathbb{R}^2$ that is equidistant from the three disks. Furthermore, v_{ijk} lies outside each of the three disks.*

In light of the above lemma, for any subset of three disks $\{D_i, D_j, D_k\} \subseteq \mathcal{D}$ that forms a strongly convex independent set, there exists a disk centered at v_{ijk} that is tangent to all three disks. We use D_{ijk} to denote this disk, and let r_{ijk} denote its radius. Note that $r_{ijk} = d_{p_i}(v_{ijk}) = d_{p_j}(v_{ijk}) = d_{p_k}(v_{ijk})$.

Since $\mathcal{D}$ is strongly convex, each disk in $\mathcal{D}$ contributes a single arc to the convex hull boundary $\partial\mathcal{H}(\mathcal{D})$, and $\partial\mathcal{H}(\mathcal{D})$ consists of an alternating sequence of disk arcs and line segments, where each line segment is an outer common tangent of two adjacent disks along $\partial\mathcal{H}(\mathcal{D})$. Let

$$\mathcal{D} = \langle D_1, D_2, \ldots, D_n\rangle$$

be a cyclic list of the disks whose arcs appear on $\partial\mathcal{H}(\mathcal{D})$, ordered counterclockwise. For any $i \neq j$, let $\mathcal{D}(i,j)$ denote the sublist of $\mathcal{D}$ from D_i to D_j in counterclockwise order along $\partial\mathcal{H}(\mathcal{D})$ excluding D_i and D_j. That is,

$$\mathcal{D}(i,j) = \begin{cases} \langle D_{i+1}, D_{i+2}, \ldots, D_{j-1}\rangle, & \text{if } i < j, \\ \langle D_{i+1}, D_{i+2}, \ldots, D_n, D_1, \ldots, D_{j-1}\rangle, & \text{if } i > j. \end{cases}$$

In addition, we have the following lemma, which will be used later (see the full paper for the proof).

▶ **Lemma 2.** *For any subset $\mathcal{D}' \subseteq \mathcal{D}$ that is strongly convex, the counterclockwise order of the disks of $\mathcal{D}'$ along $\partial\mathcal{H}(\mathcal{D}')$ is consistent with the index order of $\mathcal{D}$.*

Our algorithm is based on dynamic programming. In Section 3.1, we describe the algorithm, define the subproblems of the dynamic program, and present the dependency relations. We prove the correctness of the algorithm in Section 3.2, and discuss how to implement it efficiently in Section 3.3.

3.1 Algorithm description

For three disk indices i, j, k, we call an ordered triple (i, j, k) a *canonical triple* if D_i, D_j, and D_k appear on $\partial\mathcal{H}(\mathcal{D})$ in counterclockwise order and $\{D_i, D_j, D_k\}$ is a strongly convex independent set.

For a canonical triple (i, j, k), since $\{D_i, D_j, D_k\}$ is a strongly convex independent set, the disk D_{ijk} exists by Lemma 1. We define $\mathcal{D}_k(i,j)$ as the set of disks $D \in \mathcal{D}(i,j)$ such that D is disjoint from D_i, D_j, and D_{ijk}. For convenience, for any two disk indices i and j such that D_i is disjoint from D_j, we also treat $(i, j, 0)$ as a canonical triple, and define $\mathcal{D}_0(i,j)$ to be the set of disks $D \in \mathcal{D}(i,j)$ that are disjoint from both D_i and D_j.

For a canonical triple (i, j, k), including the case $k = 0$, let $\mathcal{D}'_k(i,j)$ denote the subset of disks $D \in \mathcal{D}_k(i,j)$ such that $\{D, D_i, D_j\}$ is strongly convex.

Subproblems. For a canonical triple (i,j,k), including the case $k=0$, we define $f(i,j,k)$ to be the maximum total weight of any subset $\mathcal{D}' \subseteq \mathcal{D}_k(i,j)$ such that $\mathcal{D}' \cup \{D_i, D_j\}$ forms a strongly convex independent set. If no such subset $\mathcal{D}'$ exists, we set $f(i,j,k)=0$.

The following lemma gives the dependency relation among the subproblems. Since the proof is lengthy and technical, we defer it to Section 3.2.

▶ **Lemma 3.** *For any canonical triple (i,j,k), including the case $k=0$, the following holds:*

$$f(i,j,k) = \begin{cases} \max\limits_{D_l \in \mathcal{D}'_k(i,j)} \big(f(i,l,j) + f(l,j,i) + W_l\big), & \text{if } \mathcal{D}'_k(i,j) \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

We define W^* to be the optimal objective value of our problem, that is, W^* equals the total weight of a maximum-weight independent set of $\mathcal{D}$. For convenience, if disks D_i and D_j intersect, we define $f(i,j,0) = -W_i - W_j$. The next lemma shows that the optimal value W^* can be computed using the values $f(i,j,0)$ for all pairs (i,j) (see the full paper for the proof). This explains our choice of $f(i,j,k)$ as the subproblems in the dynamic program.

▶ **Lemma 4.** $W^* = \max_{1 \le i,j \le n}(f(i,j,0) + W_i + W_j)$.

3.2 Algorithm correctness: Proving Lemma 3

We now prove Lemma 3. Consider a canonical triple (i,j,k). Without loss of generality, we assume that the centers p_i and p_j of disks D_i and D_j lie on the same horizontal line, with p_i to the left of p_j. This assumption allows us to clearly distinguish the upper and lower (outer) common tangents of D_i and D_j.

If $k \neq 0$, then by definition $\{D_i, D_j, D_k\}$ is a strongly convex independent set, and the disk D_{ijk} exists by Lemma 1. In the special case $k=0$, we let D_{ijk} denote the upper halfplane bounded by the upper common tangent line of D_i and D_j. Accordingly, its center v_{ijk} is interpreted as a point with infinitely large y-coordinate and its radius r_{ijk} is taken to be infinite. With this convention, the arguments below apply uniformly to both cases $k=0$ and $k \neq 0$.

If $\mathcal{D}'_k(i,j) = \emptyset$, then $f(i,j,k)=0$ by Corollary 6, which is proved using Lemma 5.

▶ **Lemma 5.** *For any subset $\mathcal{D}' \subseteq \mathcal{D}_k(i,j)$ such that $\mathcal{D}' \cup \{D_i, D_j\}$ is a strongly convex independent set, the set $\mathcal{D}'$ must contain a disk D for which $\{D, D_i, D_j\}$ is a strongly convex independent set.*

Proof. Let $S = \mathcal{D}' \cup \{D_i, D_j\}$. Since S is strongly convex, by Lemma 2, the counterclockwise order of the disks of S along $\partial\mathcal{H}(S)$ is consistent with the index order of $\mathcal{D}$. Because $\mathcal{D}' \subseteq \mathcal{D}_k(i,j) \subseteq \mathcal{D}(i,j)$, the arcs of D_i and D_j on $\partial\mathcal{H}(S)$ must be adjacent; that is, they are connected by a line segment on $\partial\mathcal{H}(S)$, and this segment is a common tangent of D_i and D_j. Since we have assumed that the centers p_i and p_j lie on a horizontal line, this segment must be the upper common tangent of D_i and D_j.

The above argument shows that the upper common tangent of D_i and D_j must be an edge of $\mathcal{H}(S)$. Since S is strongly convex, the lower common tangent of D_i and D_j cannot be an edge of $\mathcal{H}(S)$ (since otherwise it would not be possible that each disk of S has a single arc appearing on $\partial\mathcal{H}(S)$, contradicting that S is strongly convex). This implies that at least one disk $D \in \mathcal{D}'$ must have a point lying strictly below the supporting line of the lower common tangent of D_i and D_j (see Figure 3). Consequently, the set $\{D, D_i, D_j\}$ is strongly convex. Since S is an independent set, $\{D, D_i, D_j\}$ is also an independent set, completing the proof. ◀

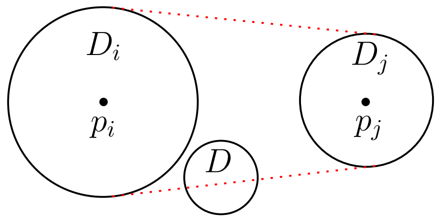

Figure 3 Illustrating the proof of Lemma 5.

▶ **Corollary 6.** *If $\mathcal{D}'_k(i,j) = \emptyset$, then $\mathcal{D}_k(i,j)$ does not contain any subset $\mathcal{D}'$ such that $\mathcal{D}' \cup \{D_i, D_j\}$ is a strongly convex independent set. Consequently, $f(i,j,k) = 0$.*

Proof. Suppose, for contradiction, that $\mathcal{D}_k(i,j)$ contains a subset $\mathcal{D}'$ such that $\mathcal{D}' \cup \{D_i, D_j\}$ is a strongly convex independent set. By Lemma 5, the set $\mathcal{D}'$ must contain a disk D such that $\{D, D_i, D_j\}$ is a strongly convex independent set. Since $\mathcal{D}' \subseteq \mathcal{D}_k(i,j)$, it follows that $\mathcal{D}_k(i,j)$ contains such a disk D, contradicting the assumption that $\mathcal{D}'_k(i,j) = \emptyset$. The corollary follows. ◀

In the following, we assume that $\mathcal{D}'_k(i,j) \neq \emptyset$. We will prove Lemma 3 in two parts: a *forward* direction and a *backward* direction.

Let S be an optimal solution defining $f(i,j,k)$; that is, $S \subseteq \mathcal{D}_k(i,j)$, the set $S \cup \{D_i, D_j\}$ is a strongly convex independent set, and $f(i,j,k) = W(S)$. In the forward direction, we show that S contains a disk $D_l \in \mathcal{D}'_k(i,j)$ such that S can be partitioned as $S = S_1 \cup S_2 \cup \{D_l\}$, where $S_1 \subseteq \mathcal{D}_j(i,l)$ and $S_2 \subseteq \mathcal{D}_i(l,j)$, and both $S_1 \cup \{D_i, D_l\}$ and $S_2 \cup \{D_l, D_j\}$ are strongly convex independent sets. This implies that $f(i,j,k) \le \max_{D_l \in \mathcal{D}'_k(i,j)} \big(f(i,l,j) + f(l,j,i) + W_l\big)$.

In the backward direction, we show that for any disk $D_l \in \mathcal{D}'_k(i,j)$ and any sets $S_1 \subseteq \mathcal{D}_j(i,l)$ and $S_2 \subseteq \mathcal{D}_i(l,j)$ such that both $S_1 \cup \{D_i, D_l\}$ and $S_2 \cup \{D_l, D_j\}$ are strongly convex independent sets, the set $S_1 \cup S_2 \cup \{D_l\}$ is contained in $\mathcal{D}_k(i,j)$ and $S_1 \cup S_2 \cup \{D_l, D_i, D_j\}$ is a strongly convex independent set. This implies that $f(i,j,k) \ge \max_{D_l \in \mathcal{D}'_k(i,j)} \big(f(i,l,j) + f(l,j,i) + W_l\big)$.

Together, the two directions establish Lemma 3.

3.2.1 The forward direction

Let S be an optimal solution defining $f(i,j,k)$; that is, $S \subseteq \mathcal{D}_k(i,j)$, the set $S \cup \{D_i, D_j\}$ is a strongly convex independent set, and $f(i,j,k) = W(S)$.

Let γ_{ij} denote the *bisector* of D_i and D_j, which consists of all points q that are equidistant from D_i and D_j. It is well known that γ_{ij} is a branch of a hyperbola (degenerating to a line when $r_i = r_j$) [29]. Moreover, since we assume that the centers of D_i and D_j lie on a horizontal line, γ_{ij} is y-monotone.

A total order $\prec$ on γ_{ij}. For any two points $a, b \in \gamma_{ij}$, we define a total order $a \prec b$ if a has a smaller y-coordinate than b. The following lemma will be frequently used.

▶ **Lemma 7.** *Suppose $D_h \in \mathcal{D}$ is a disk such that $\{D_h, D_i, D_j\}$ is a strongly convex independent set, and let v be a point on γ_{ij}. If $D_h \in \mathcal{D}(i,j)$, then $v_{ijh} \prec v$ if and only if $d_{p_i}(v) < d_{p_h}(v)$; see Figure 4. If $D_h \in \mathcal{D}(j,i)$, then $v \prec v_{ijh}$ if and only if $d_{p_i}(v) < d_{p_h}(v)$.*

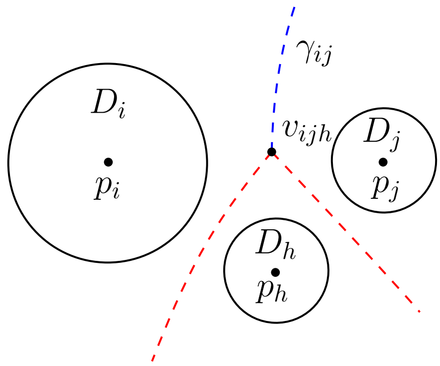

Figure 4 Illustrating Lemma 7. The three dashed curves (two red and one blue) are edges of Voronoi diagram of the three disks. The blue curve belongs to γ_{ij}.

Proof. We prove only the case where $D_h \in \mathcal{D}(i,j)$; the other case can be handled analogously.

Let $\mathcal{D}' = \{D_h, D_i, D_j\}$. Consider the additively-weighted Voronoi diagram $\mathrm{Vor}(\mathcal{D}')$ of the three weighted centers of $\mathcal{D}'$. By Lemma 1, the diagram has a unique Voronoi vertex, namely v_{ijh}. Note that $v_{ijh} \in \gamma_{ij}$. Since $\mathcal{D}'$ is strongly convex, Lemma 2 implies that D_i, D_h, and D_j appear on $\partial\mathcal{H}(\mathcal{D}')$ in counterclockwise order.

Consequently, the portion of the bisector γ_{ij} above v_{ijh} belongs to the common boundary between the Voronoi cells of p_i and p_j, while the portion of γ_{ij} below v_{ijh} lies entirely within the Voronoi cell of p_h. Therefore, for any point $v \in \gamma_{ij}$, we have $v_{ijh} \prec v$ if and only if $d_{p_i}(v) < d_{p_h}(v)$. ◀

By Lemma 5, the set S contains at least one disk D such that $\{D, D_i, D_j\}$ is strongly convex. For each disk $D_h \in S$ with this property, the point v_{ijh} exists by Lemma 1. Among all such disks $D_h \in S$, let D_l be the one for which v_{ijl} is largest under the order $\prec$. Due to the general position assumption, D_l is unique. Define $S_1 = S \cap \mathcal{D}(i,l)$ and $S_2 = S \cap \mathcal{D}(l,j)$. Since $S \subseteq \mathcal{D}_k(i,j)$, the sets S_1, S_2, and $\{D_l\}$ form a partition of S. By definition, $D_l \in \mathcal{D}'_k(i,j)$.

To complete the proof of the forward direction, it remains to show that $S_1 \subseteq \mathcal{D}_j(i,l)$, $S_2 \subseteq \mathcal{D}_i(l,j)$, and that both $S_1 \cup \{D_i, D_l\}$ and $S_2 \cup \{D_l, D_j\}$ are strongly convex independent sets. These statements are proved in the next two lemmas.

▶ **Lemma 8.** *$S_1 \subseteq \mathcal{D}_j(i,l)$ and $S_2 \subseteq \mathcal{D}_i(l,j)$.*

Proof. We only prove $S_1 \subseteq \mathcal{D}_j(i,l)$; the other case can be proved analogously.

Consider a disk $D_h \in S_1$. Our goal is to prove $D_h \in \mathcal{D}_j(i,l)$. To this end, we need to show that D_h is disjoint from the three disks D_i, D_l, and D_{ijl}. Since $S_1 \subseteq S$, $D_l \in S$, and $S \cup \{D_i, D_j\}$ is an independent set, we know that D_h is disjoint from both D_i and D_l. Hence, it remains to prove that D_h is disjoint from D_{ijl}, or equivalently, $d_{p_h}(v_{ijl}) > r_{ijl}$; recall that r_{ijl} denotes the radius of D_{ijl} and $r_{ijl} = d_{p_i}(v_{ijl})$. Hence, it suffices to show that $d_{p_h}(v_{ijl}) > d_{p_i}(v_{ijl})$. Let $S' = \{D_h, D_i, D_j\}$. Depending on whether S' is strongly convex, we distinguish two cases.

If S' is strongly convex, then by Lemma 1, v_{ijh} exists. By the definition of D_l, we have $v_{ijh} \prec v_{ijl}$. Consequently, applying Lemma 7 (with $v = v_{ijl}$) obtains $d_{p_i}(v_{ijl}) < d_{p_h}(v_{ijl})$.

We now consider the case in which S' is not strongly convex. First of all, since $\mathcal{D}$ is strongly convex and $S' \subseteq \mathcal{D}$, S' must be convex. Since $S \cup \{D_i, D_j\}$ is strongly convex and $S \subseteq \mathcal{D}(i,j)$, as argued in the proof of Lemma 5, the upper common tangent of D_i and D_j

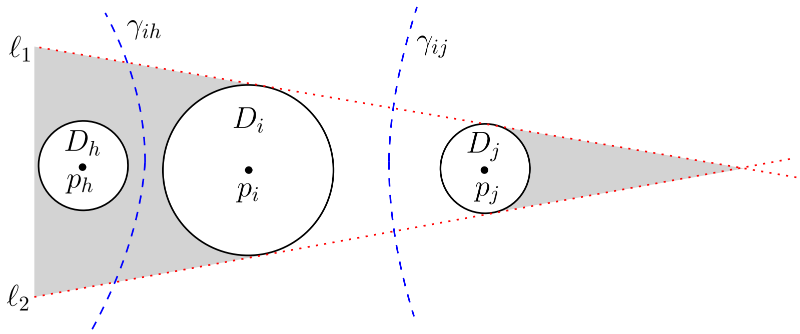

Figure 5 The disk D_h can only be in one of the two shaded regions. The curve γ_{ij} is the bisector of D_i and D_j, and γ_{ih} is the bisector of D_i and D_h. The two bisectors are the only two edges of the Voronoi diagram of the three disks.

must be on $\partial\mathcal{H}(S \cup \{D_i, D_j\})$. Since $S' \subseteq S \cup \{D_i, D_j\}$, the upper common tangent of D_i and D_j must be on $\partial\mathcal{H}(S')$. Let ℓ_1 and ℓ_2 be the upper and lower common tangent lines of D_i and D_j, respectively.

Since S' is independent and convex but not strongly convex, D_h must be either in the region to the left of D_i, below ℓ_1, and above ℓ_2, or in the region to the right of D_j, below ℓ_1, and above ℓ_2 (see Figure 5). In either case, if we consider the Voronoi diagram $\mathrm{Vor}(S')$ of the disks of S', then the bisector γ_{ij} is a common edge of the Voronoi cells of D_i and D_j. This implies that for any point $v \in \gamma_{ij}$, it holds that $d_{p_i}(v) < d_{p_h}(v)$. Since $v_{ijl} \in \gamma_{ij}$, we obtain $d_{p_i}(v_{ijl}) < d_{p_h}(v_{ijl})$. ◀

▶ **Lemma 9.** *Both $S_1 \cup \{D_i, D_l\}$ and $S_2 \cup \{D_l, D_j\}$ are strongly convex independent sets.*

Proof. See the full paper for the proof. ◀

3.2.2 The backward direction

Consider a disk $D_l \in \mathcal{D}'_k(i,j)$ together with $S_1 \subseteq \mathcal{D}_j(i,l)$ and $S_2 \subseteq \mathcal{D}_i(l,j)$ such that both $S_1 \cup \{D_i, D_l\}$ and $S_2 \cup \{D_l, D_j\}$ are strongly convex independent sets. Our goal is to show that $S_1 \cup S_2 \cup \{D_l\} \subseteq \mathcal{D}_k(i,j)$ and $S_1 \cup S_2 \cup \{D_l, D_i, D_j\}$ is a strongly convex independent set. Let $S = S_1 \cup S_2 \cup \{D_l\}$.

We begin with arguing $S \subseteq \mathcal{D}_k(i,j)$. Let D_h be a disk of S. If D_h is D_l, then since $D_l \in \mathcal{D}'_k(i,j)$ and $\mathcal{D}'_k(i,j) \subseteq \mathcal{D}_k(i,j)$, we have $D_l \in \mathcal{D}_k(i,j)$. Otherwise, D_h is either in S_1 or in S_2. We assume that $D_h \in S_1$ (the other case can be handled analogously). In the following, we argue that $D_h \in \mathcal{D}_k(i,j)$. To this end, since $D_h \in S_1 \subseteq \mathcal{D}_j(i,l) \subseteq \mathcal{D}(i,l) \subseteq \mathcal{D}(i,j)$, we need to show that D_h is disjoint from D_i, D_j, and D_{ijk}.

First of all, since $D_h \in S_1 \subseteq \mathcal{D}_j(i,l)$, D_h must be disjoint from D_i. In the next two lemmas, we prove that D_h is disjoint from D_{ijk} and D_j, respectively.

▶ **Lemma 10.** *D_h is disjoint from D_{ijk}.*

Proof. It suffices to show that $d_{p_h}(v_{ijk}) > r_{ijk}$, or equivalently, $d_{p_h}(v_{ijk}) > d_{p_i}(v_{ijk})$ as $r_{ijk} = d_{p_i}(v_{ijk})$.

Since $D_l \in \mathcal{D}'_k(i,j)$, $\{D_l, D_i, D_j\}$ is a strongly convex independent set and thus v_{ijl} exists by Lemma 1. Clearly, both v_{ijk} and v_{ijl} are on the bisector γ_{ij} of D_i and D_j. Depending on whether $\{D_i, D_j, D_h\}$ is strongly convex, we distinguish two cases.

- If $\{D_i, D_j, D_h\}$ is not strongly convex, then similarly to the proof of Lemma 8, we can obtain that for any point $v \in \gamma_{ij}$, $d_{p_i}(v) < d_{p_h}(v)$. As $v_{ijk} \in \gamma_{ij}$, it follows that $d_{p_i}(v_{ijk}) < d_{p_h}(v_{ijk})$.
- If $\{D_i, D_j, D_h\}$ is strongly convex, then v_{ijh} exists by Lemma 1. We claim that $v_{ijh} \prec v_{ijl}$. To see this, since $D_h \in S_1 \subseteq \mathcal{D}_j(i,l)$, D_h is disjoint from D_{ijl}, or equivalently, $d_{p_i}(v_{ijl}) < d_{p_h}(v_{ijl})$. By Lemma 7 (with $v = v_{ijl}$), we obtain $v_{ijh} \prec v_{ijl}$.
  Since $D_l \in \mathcal{D}_k(i,j)$, following the similar argument as above, we can obtain $v_{ijl} \prec v_{ijk}$. Hence, we have $v_{ijh} \prec v_{ijk}$. Consequently, applying Lemma 7 again (with $v = v_{ijk}$) leads to $d_{p_i}(v_{ijk}) < d_{p_h}(v_{ijk})$.

The lemma thus follows. ◀

▶ **Lemma 11.** *D_h is disjoint from D_j.*

Proof. See the full paper for the proof. ◀

The above proves that $S \subseteq \mathcal{D}_k(i,j)$. The following lemma finally proves that $S \cup \{D_i, D_j\}$ is a strongly convex independent set (see the full paper for the proof). This completes the proof of the backward direction.

▶ **Lemma 12.** *$S \cup \{D_i, D_j\}$ is a strongly convex independent set.*

3.3 Algorithm implementation

We now discuss how to implement the algorithm. We focus on computing W^*. By slightly modifying the algorithm using a standard backtracking technique, an actual maximum-weight independent set can also be recovered within asymptotically the same time complexity.

To compute W^*, Lemma 4 implies that it suffices to compute $f(i,j,k)$ for all canonical triples (i,j,k), including the case $k = 0$. To this end, by Equation (1), we must determine an order for solving the subproblems such that, when computing a value $f(i,j,k)$, all values $f(i,l,j)$ and $f(l,j,i)$ for $D_l \in \mathcal{D}'_k(i,j)$ have already been computed.

We process pairs (i,j) in increasing order of j, and for each fixed j in decreasing order of i: for $j = 2, \ldots, n$, we consider $i = j-1, j-2, \ldots, 1$ in this order. If $D_i \cap D_j \neq \emptyset$, we simply set $f(i,j,0) = -W_i - W_j$. Otherwise, we compute $f(i,j,0)$ using Equation (1). Next, for each disk $D_k \in \mathcal{D}(j,i)$ such that (i,j,k) is canonical, we compute $f(i,j,k)$ using Equation (1). With this ordering, every f-value appearing on the right-hand side of Equation (1) involves indices that lie between i and j in cyclic order, and hence has already been computed.

Using Equation (1), each subproblem $f(i,j,k)$ can be computed in $O(n)$ time by scanning all disks $D_l \in \mathcal{D}'_k(i,j)$. Since there are $O(n^3)$ subproblems, the total running time of the algorithm is $O(n^4)$.

An improved solution. We now improve the running time to $O(n^3 \log n)$. More specifically, we show that for any fixed pair (i,j), the $O(n)$ subproblems $f(i,j,k)$ for all relevant indices k can be computed in a total of $O(n \log n)$ time. This improvement is achieved by exploiting a monotonicity property of the sets $\mathcal{D}'_k(i,j)$ with respect to a suitable ordering of the indices k.

Consider a pair (i,j) such that D_i does not intersect D_j, and let (i,j,k) be a canonical triple. For each disk $D_l \in \mathcal{D}'_k(i,j)$, we define its *cost* as

$$cost(D_l) = f(i,l,j) + f(l,j,i) + W_l.$$

Define $\mathcal{D}'(i,j)$ as the set of disks $D \in \mathcal{D}(i,j)$ such that $\{D, D_i, D_j\}$ forms a strongly convex independent set. Then $\mathcal{D}'_k(i,j)$ consists precisely of those disks in $\mathcal{D}'(i,j)$ that are disjoint from the disk D_{ijk}. By Equation (1), computing $f(i,j,k)$ reduces to finding the disk of maximum cost among all disks in $\mathcal{D}'(i,j)$ that are disjoint from D_{ijk}.

We process a fixed pair (i,j) using the following approach, which computes $f(i,j,k)$ for all $k \in K$, where K denotes the set of indices k such that (i,j,k) is a canonical triple. The total running time of this procedure is $O(n \log n)$.

Without loss of generality, we assume that the centers p_i and p_j lie on the same horizontal line, with p_i to the left of p_j. We first compute the set K and the set $\mathcal{D}'(i,j)$, which can be done in $O(n)$ time.

For each $k \in K$, since (i,j,k) is a canonical triple, the set $\{D_i, D_j, D_k\}$ is strongly convex, and thus the vertex v_{ijk} exists by Lemma 1. The following lemma, which establishes a monotonicity property of the sets $\mathcal{D}'_k(i,j)$, is central to our approach.

▶ **Lemma 13.** *Consider any $k, k' \in K$ with $v_{ijk} \prec v_{ijk'}$. For any disk $D_h \in \mathcal{D}'(i,j)$, if D_h is disjoint from D_{ijk}, then D_h is also disjoint from $D_{ijk'}$. Consequently, $\mathcal{D}'_k(i,j) \subseteq \mathcal{D}'_{k'}(i,j)$.*

Proof. Consider a disk $D_h \in \mathcal{D}'(i,j)$ such that D_h is disjoint from D_{ijk}. Our goal is to show that D_h is also disjoint from $D_{ijk'}$, or equivalently, $d_{p_h}(v_{ijk'}) > r_{ijk'}$.

Since D_h is disjoint from D_{ijk}, we have $d_{p_h}(v_{ijk}) > r_{ijk}$. As $D_h \in \mathcal{D}'(i,j)$, $\{D_h, D_i, D_j\}$ is a strongly convex independent set. Therefore, v_{ijh} exists by Lemma 1. Since $d_{p_h}(v_{ijk}) > r_{ijk} = d_{p_i}(v_{ijk})$, applying Lemma 7 (with $v = v_{ijk}$) gives $v_{ijh} \prec v_{ijk}$.

On the other hand, since $v_{ijk} \prec v_{ijk'}$, we have $v_{ijh} \prec v_{ijk'}$. Consequently, applying Lemma 7 (with $v = v_{ijk'}$) gives $d_{p_h}(v_{ijk'}) > d_{p_i}(v_{ijk'}) = r_{ijk'}$. This proves the lemma. ◀

In light of Lemma 13, our algorithm proceeds as follows. We sort all indices $k \in K$ according to the $\prec$ order on γ_{ij}; let $k_1, k_2, \ldots, k_g$ be the resulting sequence, where $g = |K|$. Since $g \leq n$, this sorting step takes $O(n \log n)$ time. Next, for each disk $D_h \in \mathcal{D}'(i,j)$, we compute the smallest index $t \in [1,g]$ such that D_h is disjoint from D_{ijk_t}. By Lemma 13, this can be done in $O(\log n)$ time via binary search on the sequence $k_1, k_2, \ldots, k_g$. We then assign this value t as a *label* to D_h.

Recall that our goal is to compute $f(i,j,k_t)$ for all $t = 1, 2, \ldots, g$, where each $f(i,j,k_t)$ is equal to the maximum cost among all disks disjoint from D_{ijk_t}. We process the indices in increasing order of t. For convenience, let $f(i,j,k_0) = 0$. For each $t \geq 1$, we initialize $f(i,j,k_t) = f(i,j,k_{t-1})$, and then examine all disks $D_l \in \mathcal{D}'(i,j)$ whose label is exactly t. For each such disk, if $f(i,j,k_t) < \mathit{cost}(D_l)$, we update $f(i,j,k_t) = \mathit{cost}(D_l)$. In this manner, all values $f(i,j,k_t)$ for $t = 1, 2, \ldots, g$ can be computed in a total of $O(n)$ time.

This completes the processing for a fixed pair (i,j), with a total running time of $O(n \log n)$. Since there are $O(n^2)$ choices of pairs (i,j), the overall running time of the algorithm is $O(n^3 \log n)$. The following lemma summarizes our result.

▶ **Lemma 14.** *Given a set $\mathcal{D}$ of n weighted disks in the plane such that the disks are in strongly convex position and $\mathcal{D}$ admits a maximum-weight independent set that is strongly convex, a maximum-weight independent set of $\mathcal{D}$ can be computed in $O(n^3 \log n)$ time.*

4 The strongly-convex-input case

In this section, we consider the strongly-convex-input case, in which the input disks of $\mathcal{D}$ are in strongly convex position but $\mathcal{D}$ does not necessarily admit a maximum-weight independent set that is strongly convex (see Figure 6).

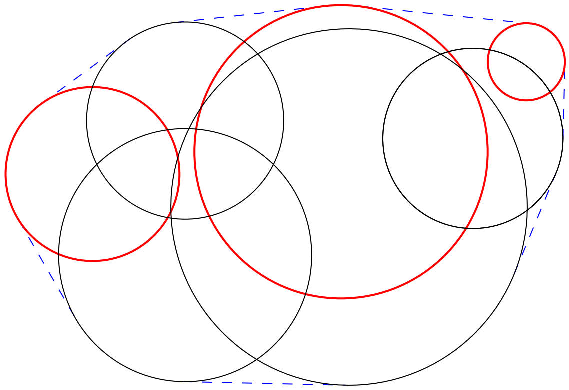

Figure 6 Illustrating an example where the input set of disks is strongly convex, but the only maximum-weight independent set (shown in red) is not strongly convex (assuming the weight of each disk is 1).

To handle this case, we reduce it to the setting in Section 3. To this end, we construct a set B of $O(n)$ *auxiliary points* satisfying the following two properties:

1. No point of B is contained in any disk of $\mathcal{D}$.
2. For any subset $\mathcal{D}' \subseteq \mathcal{D}$, $\mathcal{D}' \cup B$ is strongly convex (in particular, $\mathcal{D} \cup B$ is strongly convex).

Each point of B is treated as a special disk and assigned a weight of 1. Let $S = \mathcal{D} \cup B$. By the first property, any maximum-weight independent set of S must include all points of B. Therefore, if S' is a maximum-weight independent set of S, then $S' \setminus B$ is a maximum-weight independent set of $\mathcal{D}$. In this way, computing a maximum-weight independent set of $\mathcal{D}$ reduces to computing a maximum-weight independent set of S. Moreover, by the second property, S is strongly convex and admits a maximum-weight independent set that is strongly convex. Consequently, computing a maximum-weight independent set of S becomes an instance of the case in Section 3 and can be solved in $O(n^3 \log n)$ time by Lemma 14.

In the remainder of this section, we focus on constructing a set B of $O(n)$ auxiliary points satisfying the above two properties, and show that such a set can be computed in $O(n \log n)$ time.

To simplify the discussion, we assume that each disk in $\mathcal{D}$ has positive radius. The same approach extends to the case where some disks degenerate to points, although the exposition becomes more involved. We also assume that $n \geq 3$, since smaller instances can be handled directly in $O(1)$ time. Because $\mathcal{D}$ is strongly convex, each disk contributes exactly one arc to the boundary $\partial\mathcal{H}(\mathcal{D})$. As in Section 3, let $\mathcal{D} = \langle D_1, \ldots, D_n \rangle$ denote the cyclic order of disks appearing on $\partial\mathcal{H}(\mathcal{D})$ in counterclockwise order.

4.1 Defining *B*

For each disk $D_i \in \mathcal{D}$, D_i contributes a single arc α_i to the boundary $\partial\mathcal{H}(\mathcal{D})$. Let a_i denote the midpoint of α_i, and let ℓ_i be the line through a_i that is tangent to D_i. Thus, ℓ_i is also tangent to $\mathcal{H}(\mathcal{D})$ at a_i. We orient ℓ_i so that D_i lies to the left of ℓ_i (see Figure 7).

For each pair of consecutive disks D_i and D_{i+1} along $\partial\mathcal{H}(\mathcal{D})$ (if $i = n$, then we let $i + 1$ take modulo n, which equals 1), we proceed as follows. Let $\rho_{i,i+1}$ be the directed common tangent line of D_i and D_{i+1} such that both disks lie to the left of $\rho_{i,i+1}$. By the definition of the disk order along $\partial\mathcal{H}(\mathcal{D})$, the convex hull $\mathcal{H}(\mathcal{D})$ lies entirely to the left of $\rho_{i,i+1}$, and the segment of $\rho_{i,i+1}$ between the two tangency points on D_i and D_{i+1} forms an edge of $\mathcal{H}(\mathcal{D})$.

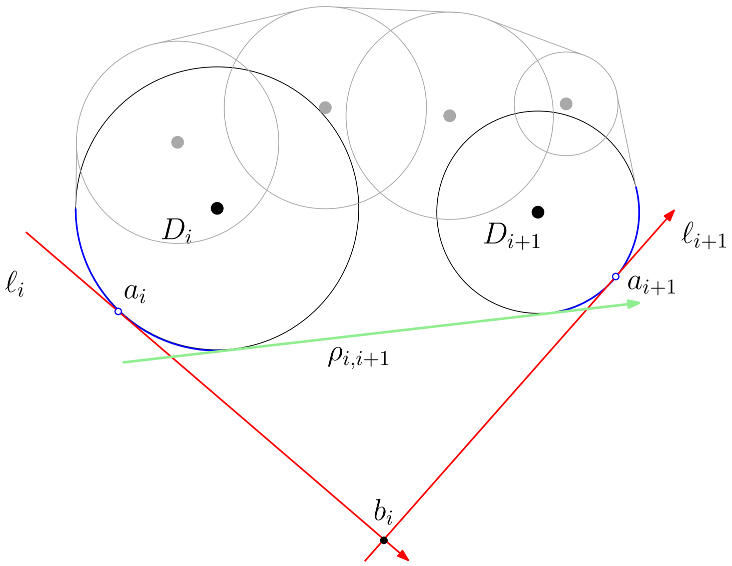

Figure 7 Illustrating the definition of b_i.

Since ℓ_i is tangent to D_i at the midpoint of α_i, the line ℓ_i must intersect $\rho_{i,i+1}$. Similarly, ℓ_{i+1} intersects $\rho_{i,i+1}$. Moreover, because $n \geq 3$, the lines ℓ_i and ℓ_{i+1} intersect at a point b_i that lies to the right of $\rho_{i,i+1}$ (see Figure 7).

We define B as the set of all such intersection points b_i, for $1 \leq i \leq n$. Thus, $|B| = n$.

If we treat each point b_i as a special disk, then this construction places b_{i-1}, D_i, and b_i in a degenerate configuration, since ℓ_i is tangent to all three. Note that our algorithm in Section 3 also works for such degenerate cases. Alternatively, if desired, one could perturb each point b_i infinitesimally toward $\rho_{i,i+1}$ so that the set $S = \mathcal{D} \cup B$ is in general position. In what follows, we retain the definition of B above and allow degeneracies, as this simplifies the exposition.

Since the convex hull $\mathcal{H}(\mathcal{D})$ can be computed in $O(n \log n)$ time [26], the set B can also be constructed in $O(n \log n)$ time.

The following lemma proves that B has the two desired properties.

▶ **Lemma 15.** *The set B has the two desired properties.*

Proof. See the full paper for the proof. ◀

For reference purpose, the following lemma summarizes our result in this section.

▶ **Lemma 16.** *Given a set $\mathcal{D}$ of n weighted disks in strongly convex position in the plane, a maximum-weight independent set of $\mathcal{D}$ can be computed in $O(n^3 \log n)$ time.*

5 The general case

In this section, we finally consider the general case in which $\mathcal{D}$ is not necessarily strongly convex (but is still convex). In particular, a disk may contribute more than one (maximal) arc to $\partial\mathcal{H}(\mathcal{D})$.

To solve the problem, we reduce it to the case in Section 4. To this end, we create a set Z of $O(n)$ *auxiliary points* as special disks with each disk assigned a weight equal to 1 such that the following two properties hold:

1. No point of Z is contained in any disk of $\mathcal{D}$.
2. $\mathcal{D} \cup Z$ is strongly convex.

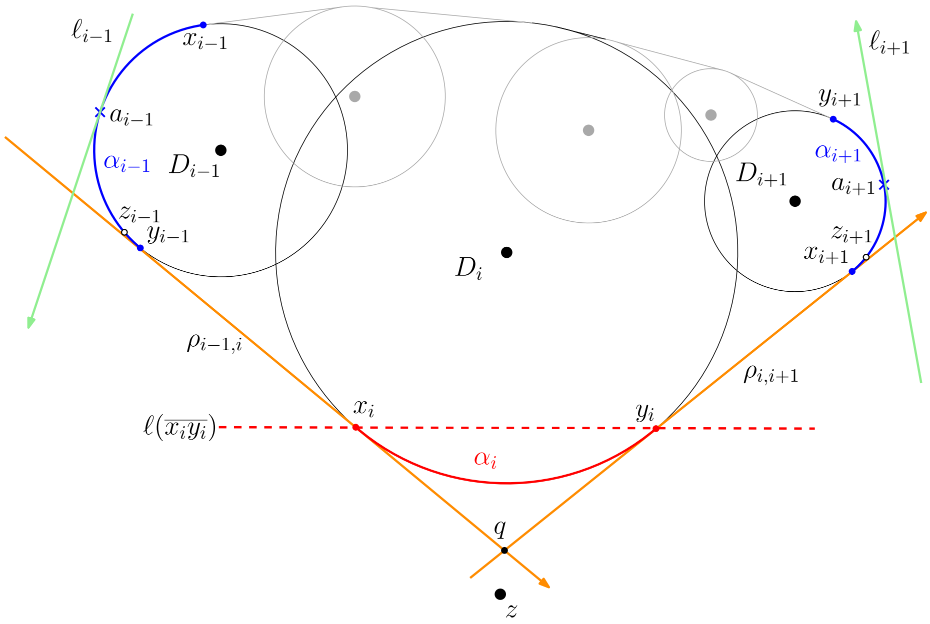

Figure 8 Illustration of the definition of z.

Let $S = \mathcal{D} \cup Z$. As in the reduction in Section 4, due to the first property, it suffices to compute a maximum-weight independent set in S. This is an instance of the problem in Section 4 due to the second property, and thus the problem can be solved in $O(n^3 \log n)$ time by Lemma 16.

In the rest of this section, we will focus on defining a set Z of $O(n)$ points satisfying the above two properties. We will also show that computing Z can be done in $O(n^2 \log n)$ time.

Consider a disk D_i that has more than one arc on $\partial\mathcal{H}(\mathcal{D})$. Note that at most one such arc can have a radian measure larger than or equal to π, and all other arcs have median measures strictly smaller than π. Let α_i be an arc of D_i on $\partial\mathcal{H}(\mathcal{D})$ whose radian measure is less than π.

Let α_{i-1} and α_{i+1} be the two disk arcs on $\mathcal{H}(\mathcal{D})$ adjacent to α_i clockwise and counterclockwise, respectively; let D_{i-1} and D_{i+1} be the disks where these arcs lie on, respectively (see Figure 8). Note that it is possible that D_{i-1} (resp., D_{i+1}) is a single point. We first assume that both D_{i-1} and D_{i+1} have radii larger than 0. We will discuss the other case later, which can be handled similarly. Refer to Figure 8 for an illustration of the notation introduced below.

For each $j \in \{i-1, i, i+1\}$, let x_j and y_j be the clockwise and counterclockwise endpoints of α_j, respectively. By definition, $\overline{y_{i-1}x_i}$ is an edge of $\mathcal{H}(\mathcal{D})$ and its supporting line is tangent to D_{i-1} at y_{i-1} and tangent to D_i at x_i. Let $\rho_{i-1,i}$ denote the supporting line of $\overline{y_{i-1}x_i}$. We orient $\rho_{i-1,i}$ so that $\mathcal{H}(\mathcal{D})$ is on the left side of $\rho_{i-1,i}$. Similarly, let $\rho_{i,i+1}$ be the oriented supporting line of $\overline{y_ix_{i+1}}$ so that $\mathcal{H}(\mathcal{D})$ is on its left side.

Let $\ell(\overline{x_iy_i})$ denote the line containing x_i and y_i. Without loss of generality, we assume that $\ell(\overline{x_iy_i})$ is horizontal such that x_i is left of y_i. Thus, the arc α_i is below $\ell(\overline{x_iy_i})$. Since the radian measure of α_i is less than π, $\rho_{i-1,i}$ must intersect $\rho_{i,i+1}$ at a point q strictly below $\ell(\overline{x_iy_i})$.

Let a_{i-1} be the middle point of the arc α_{i-1}. Let ℓ_{i-1} be an oriented line tangent to D_{i-1} at a_{i-1} such that $\mathcal{H}(\mathcal{D})$ is on the left of ℓ_{i-1}. Similarly, we define a_{i+1} and ℓ_{i+1} for D_{i+1}.

Note that by definition, q must be to the left of both ℓ_{i-1} and ℓ_{i+1}. Let q' be a point strictly below q but infinitesimally close to q. Then, q' is strictly to the right of both $\rho_{i-1,i}$ and $\rho_{i,i+1}$, and also strictly to the left of both ℓ_{i-1} and ℓ_{i+1}. This means that the region Q is not empty, where Q is the common intersection of the following four open halfplanes: the open halfplane to the right of $\rho_{i-1,i}$, the open halfplane to the right of $\rho_{i,i+1}$, the open halfplane to the left of ℓ_{i-1}, and the open halfplane to the left of ℓ_{i+1}.

We pick an arbitrary point z from Q and use it as an auxiliary point; we show below that z can be used to eliminate the arc α_i.

Let z_{i-1} be a point on ∂D_{i-1} such that $\overline{z_{i-1}z}$ is tangent to D_{i-1} at z_{i-1} and D_{i-1} is to the left of the directed line containing $\overline{z_{i-1}z}$ with the direction from z_{i-1} to z. Since z is strictly to the left of ℓ_{i-1} and strictly to the right of $\rho_{i-1,i}$, the tangent point z_{i-1} must be in the interior of the arc $\partial_{[a_{i-1},y_{i-1}]}D_{i-1}$, which is a subarc of α_{i-1}.

Symmetrically, let z_{i+1} be a point on ∂D_{i+1} such that $\overline{z_{i+1}z}$ is tangent to D_{i+1} at z_{i+1} and D_{i+1} is to the right of the directed line containing $\overline{z_{i+1}z}$ with the direction from z_{i+1} to z. Since z is strictly to the left of ℓ_{i+1} and strictly to the right of $\rho_{i,i+1}$, the tangent point z_{i+1} must be in the interior of the arc $\partial_{[x_{i+1},a_{i+1}]}D_{i+1}$, which is a subarc of α_{i+1}.

Consider the convex hull $\mathcal{H}(\mathcal{D}\cup\{z\})$. Its boundary can be obtained from $\partial\mathcal{H}(\mathcal{D})$ by replacing $\partial_{[z_{i-1},z_{i+1}]}\mathcal{H}(\mathcal{D})$ with $\overline{z_{i-1}z}\cup\overline{zz_{i+1}}$. In particular, no point of α_i appears on $\partial\mathcal{H}(\mathcal{D}\cup\{z\})$. Hence, every disk of $\mathcal{D}$ has the same number of arcs appearing in $\partial\mathcal{H}(\mathcal{D}\cup\{z\})$ as in $\partial\mathcal{H}(\mathcal{D})$ except that D_i has one less arc than before. Furthermore, since z is vertex of $\mathcal{H}(\mathcal{D}\cup\{z\})$, $\mathcal{D}\cup\{z\}$ is still convex.

The above assumes that both D_{i-1} and D_{i+1} are disks of radii larger than zero. The other case can be handled similarly. For example, if D_{i-1} is a point p_{i-1}, then following the same way as above, we have $\alpha_{i-1}=x_{i-1}=y_{i-1}=a_{i-1}=z_{i-1}=p_{i-1}$. Hence, p_{i-1} is still on $\partial\mathcal{H}(\mathcal{D}\cup\{z\})$. Therefore, it is still the case that $\mathcal{D}\cup\{z\}$ is convex and every disk of $\mathcal{D}$ has the same number of arcs appearing in $\partial\mathcal{H}(\mathcal{D}\cup\{z\})$ as in $\partial\mathcal{H}(\mathcal{D})$ except that D_i has one less arc than before. If D_{i+1} is a point, we can handle the situation similarly.

In summary, we can find a point z such that $\mathcal{D}\cup\{z\}$ is convex and every disk of $\mathcal{D}$ has the same number of arcs appearing in $\partial\mathcal{H}(\mathcal{D}\cup\{z\})$ as in $\partial\mathcal{H}(\mathcal{D})$ except that D_i has one less arc than before. As $\mathcal{H}(\mathcal{D})$ can be computed in $O(n\log n)$ time [26], such a point z can be found in $O(n\log n)$ time. We add z to Z. Note that z is outside every disk of $\mathcal{D}$.

If $\mathcal{D}\cup\{z\}$ is not strongly convex, then by following the same method we can find another auxiliary point with respect to $\mathcal{D}\cup\{z\}$ to eliminate another arc. Since $\partial\mathcal{H}(\mathcal{D})$ has $O(n)$ disk arcs [26], we can find a set Z of $O(n)$ auxiliary points such that $\mathcal{D}\cup Z$ is strongly convex (and no point of Z is inside any disk of $\mathcal{D}$). As computing each auxiliary point can be done in $O(n\log n)$ time, computing Z takes $O(n^2\log n)$ time in total (it may be possible to reduce the time to $O(n\log n)$ with a more careful implementation, but $O(n^2\log n)$ time suffices for our purpose).

The following theorem summarizes our result.

▶ **Theorem 17.** *Given a set $\mathcal{D}$ of n weighted disks in convex position in the plane, a maximum-weight independent set of $\mathcal{D}$ can be computed in $O(n^3\log n)$ time.*

The following corollary will be used in Section 6 to solve a dispersion problem.

▶ **Corollary 18.** *Given a set $\mathcal{D}$ of n weighted disks in convex position in the plane and a parameter $r>0$, one can compute in $O(n^3\log n)$ time a maximum-cardinality subset $\mathcal{D}'\subseteq\mathcal{D}$ such that the distance between any two disks in $\mathcal{D}'$ is greater than r.*

Proof. For each disk $D\in\mathcal{D}$, let $D(r)$ be the disk with the same center as D and with radius $r/2$ larger than that of D. For any subset $\mathcal{D}'\subseteq\mathcal{D}$, define $S(\mathcal{D}')=\{D(r):D\in\mathcal{D}'\}$. It is not difficult to see that for any subset $\mathcal{D}'\subseteq\mathcal{D}$, the distance between any two disks in $\mathcal{D}'$ is greater than r if and only if $S(\mathcal{D}')$ is an independent set. Hence, the problem reduces to finding a maximum-cardinality independent set from $S(\mathcal{D})$. Since $\mathcal{D}$ is in convex position, $S(\mathcal{D})$ must also be in convex position because each disk of $\mathcal{D}$ grows by the same radius to obtain $S(\mathcal{D})$. Therefore, if we assign each disk of $S(\mathcal{D})$ a weight equal to 1, then we can compute a maximum-cardinality independent set of $S(\mathcal{D})$ in $O(n^3 \log n)$ time by Theorem 17. ◀

6 The dispersion problem

Let $\mathcal{D}$ be a set of n disks in convex position in the plane and let $k > 0$ be an integer. For any two disks $D, D' \in \mathcal{D}$, we define the *disk distance* $\text{dist}(D, D')$ as the minimum distance between any two points $q \in D$ and $q' \in D'$. The *dispersion problem* for $\mathcal{D}$ asks for a subset $\mathcal{D}' \subseteq \mathcal{D}$ of k disks maximizing the minimum pairwise distance $\text{dist}(D, D')$ of any two disks $D, D' \in \mathcal{D}'$.

Let r^* denote the optimal objective value. It is not difficult to see that r^* is equal to $\text{dist}(D, D')$ for two disks $D, D' \in \mathcal{D}$. Hence, r^* belongs to the set $R = \{\text{dist}(D, D') : D, D' \in \mathcal{D}\}$. Clearly, $|R| = O(n^2)$.

Given a number $r \geq 0$, the *decision problem* is to determine whether $r < r^*$, or equivalently, whether $\mathcal{D}$ has a subset of k disks whose minimum pairwise distance is greater than r. By Corollary 18, the decision problem can be solved in $O(n^3 \log n)$ time. To find r^*, we first compute R and then do binary search in R using the decision algorithm. This yields an $O(n^3 \log^2 n)$-time algorithm to compute r^*. We can also find an optimal subset by Corollary 18, as explained in the proof of the following theorem.

▶ **Theorem 19.** *Given a set of n disks in convex position in the plane and an integer k, one can find a subset of k disks maximizing their minimum pairwise distance in $O(n^3 \log^2 n)$ time.*

Proof. We first compute r^* as discussed above. Once r^* is computed, we apply the algorithm of Corollary 18 to produce an optimal subset with $r = r^*$. The algorithm of Corollary 18 will first compute a set $S(\mathcal{D})$ of disks and then apply the algorithm of Theorem 17 on $S(\mathcal{D})$. Here we need to slightly modify the algorithm of Theorem 17 by treating each disk of $S(\mathcal{D})$ as an open disk. For example, two disks are considered to be disjoint if they are outer tangent to each other. ◀

References

1. Pankaj K. Agarwal and Nabil H. Mustafa. Independent set of intersection graphs of convex objects in 2D. *Computational Geometry: Theory and Applications*, 34(2):83–95, 2006. doi:10.1016/j.comgeo.2005.12.001.
2. Pankaj K. Agarwal, Mark H. Overmars, and Micha Sharir. Computing maximally separated sets in the plane. *SIAM Journal on Computing*, 36(3):815–834, 2006. doi:10.1137/S0097539704446591.
3. Pankaj K. Agarwal, Marc J. van Kreveld, and Subhash Suri. Label placement by maximum independent set in rectangles. *Computational Geometry: Theory and Applications*, 11:209–218, 1998. doi:10.1016/S0925-7721(98)00028-5.
4. Alok Aggarwal, Leonidas J. Guibas, James Saxe, and Peter W. Shor. A linear-time algorithm for computing the Voronoi diagram of a convex polygon. *Discrete and Computational Geometry*, 4:591–604, 1989. doi:10.1007/BF02187749.
5. Alberto Apostolico, Mikhail J. Atallah, and Susanne E. Hambrusch. New clique and independent set algorithms for circle graphs. *Discrete Applied Mathematics*, 36(1):1–24, 1992. doi:10.1016/0166-218X(92)90200-T.

6 Paul Balister, Béla Bollobás, Amites Sarkar, and Mark Walters. Connectivity of random k-nearest-neighbour graphs. *Advances in Applied Probability*, 37(1):1–24, 2005. doi:10.1239/aap/1113402397.

7 Prosenjit Bose, Paz Carmi, J Mark Keil, Anil Maheshwari, Saeed Mehrabi, Debajyoti Mondal, and Michiel Smid. Computing maximum independent set on outerstring graphs and their relatives. *Computational Geometry: Theory and Applications*, 103:101852, 2022. doi:10.1016/j.comgeo.2021.101852.

8 Onur Çağı rıcıand Bodhayan Roy. Maximum clique of disks in convex position. In *Abstracts of Computational Geometry: Young Researchers Forum (CG:YRF)*, pages 21:1–21:3, 2018. URL: https://www.computational-geometry.org/old/YRF/cgyrf2018.pdf.

9 Timothy M. Chan. Approximation algorithms for maximum independent set of pseudo-disks. *Discrete and Computational Geometry*, 48:373–392, 2012. doi:10.1007/s00454-012-9417-5.

10 Bernard Chazelle, Herbert Edelsbrunner, Michelangelo Grigni, Leonidas Guibas, Micha Sharir, and Emo Welzl. Improved bounds on weak ϵ-nets for convex sets. In *Proceedings of the 25th Annual ACM Symposium on Theory of Computing (STOC)*, pages 495–504, 1993. doi:10.1145/167088.167222.

11 Jongmin Choi, Jaegun Lee, and Hee-Kap Ahn. Efficient k-center algorithms for planar points in convex position. In *Proceedings of the 18th Algorithms and Data Structures Symposium (WADS)*, pages 262–274, 2023. doi:10.1007/978-3-031-38906-1_18.

12 Brent N. Clark, Charles J. Colbourn, and David S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86:165–177, 1990. doi:10.1016/0012-365X(90)90358-O.

13 Gautam K. Das, Guilherme D. da Fonseca, and Ramesh K. Jallu. Efficient independent set approximation in unit disk graphs. *Discrete Applied Mathematics*, 280:63–70, 2020. doi:10.1016/j.dam.2018.05.049.

14 Gautam K. Das, Minati De, Sudeshna Kolay, Subhas C. Nandy, and Susmita Sur-Kolay. Approximation algorithms for maximum independent set of a unit disk graph. *Information Processing Letters*, 115:439–446, 2015. doi:10.1016/j.ipl.2014.11.002.

15 Stefan Felsner, Rudolf Müller, and Lorenz Wernisch. Trapezoid graphs and generalizations, geometry and algorithms. *Discrete Applied Mathematics*, 74(1):13–32, 1997. doi:10.1016/S0166-218X(96)00013-3.

16 Fănică Gavril. Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph. *SIAM Journal on Computing*, 1(2):180–187, 1972. doi:10.1137/0201013.

17 Fănică Gavril. Algorithms for a maximum clique and a maximum independent set of a circle graph. *Networks*, 3(3):261–273, 1973. doi:10.1002/net.3230030305.

18 Luis H. Herrera and Pablo Pérez-Lantero. On the intersection graph of the disks with diameters the sides of a convex n-gon. *Applied Mathematics and Computation*, 411:126472, 2021. doi:10.1016/j.amc.2021.126472.

19 Clemens Huemer and Pablo Pérez-Lantero. The intersection graph of the disks with diameters the sides of a convex n-gon. *Discrete Mathematics*, 343(2):111669, 2020. doi:10.1016/j.disc.2019.111669.

20 Richard M. Karp. Reducibility among combinatorial problems. *Complexity of Computer Computations*, pages 85–103, 1972. doi:10.1007/978-1-4684-2001-2_9.

21 Andrzej Lingas. On approximation behavior and implementation of the greedy triangulation for convex planar point sets. In *Proceedings of the 2nd Annual Symposium on Computational Geometry (SoCG)*, pages 72–79, 1986. doi:10.1145/10515.10523.

22 Madhav V. Marathe, Heinz Breu, Harry B. Hunt III, Sekharipuram S. Ravi, and Daniel J. Rosenkrantz. Simple heuristics for unit disk graphs. *Networks*, 25:59–68, 1995. doi:10.1002/net.3230250205.

23 Tomomi Matsui. Approximation algorithms for maximum independent set problems and fractional coloring problems on unit disk graphs. In *Proceedings of the 2nd Japanese Conference on Discrete and Computational Geometry (JCDCG)*, pages 194–200, 1998. doi:10.1007/978-3-540-46515-7_16.

24 Charles E. Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In *Proceedings of the Conference on Communications Architectures, Protocols and Applications (SIGCOMM)*, pages 234–244, 1994. doi:10.1145/190809.190336.

25 Charles E. Perkins and Pravin Bhagwat. Ad-hoc on-demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, pages 90–100, 1999. doi:10.1109/MCSA.1999.749281.

26 David Rappaport. A convex hull algorithm for discs, and applications. *Computational Geometry: Theory and Applications*, 1(3):171–187, 1992. doi:10.1016/0925-7721(92)90015-K.

27 Dana S. Richards and Jeffrey S. Salowe. A rectilinear steiner minimal tree algorithm for convex point sets. In *Proceedings of the 2nd Scandinavian Workshop on Algorithm Theory (SWAT)*, pages 201–212, 1990. doi:10.1007/3-540-52846-6_90.

28 Paweł Rzążewski and Bartosz Walczak. Polynomial-time recognition and maximum independent set in burling graphs. In *Proceedings of the International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, pages 445–460, 2026. doi:10.1007/978-3-032-11835-6_32.

29 Micha Sharir. Intersection and closest-pair problems for a set of planar discs. *SIAM Journal on Computing*, 14(2):448–468, 1985. doi:10.1137/0214034.

30 Vishwanath R. Singireddy, Manjanna Basappa, and Joseph S.B. Mitchell. Algorithms for k-dispersion for points in convex position in the plane. In *Proceedings of the 9th International Conference on Algorithms and Discrete Applied Mathematics (CALDAM)*, pages 59–70, 2023. doi:10.1007/978-3-031-25211-2_5.

31 Anastasiia Tkachenko and Haitao Wang. Computing dominating sets in disk graphs with centers in convex position. In *Proceedings of the 17th Latin American Theoretical Informatics Symposium (LATIN)*, page to appear, 2025. doi:10.48550/arXiv.2601.22609.

32 Anastasiia Tkachenko and Haitao Wang. Computing maximum cliques in unit disk graphs. In *Proceedings of the 37th Canadian Conference on Computational Geometry (CCCG)*, pages 283–291, 2025. URL: https://cccg-wads-2025.eecs.yorku.ca/cccg-papers/6B2.pdf.

33 Anastasiia Tkachenko and Haitao Wang. Dominating set, independent set, discrete k-center, dispersion, and related problems for planar points in convex position. In *Proceedings of the 42nd International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 73:1–73:20, 2025. doi:10.4230/LIPIcs.STACS.2025.73.

34 Da-Wei Wang and Yue-Sun Kuo. A study on two geometric location problems. *Information processing letters*, 28(6):281–286, 1988. doi:10.1016/0020-0190(88)90174-3.