Maximum Clique in Disk-Like Intersection Graphs

Édouard Bonnet

Univ Lyon, CNRS, ENS de Lyon, Université Claude Bernard Lyon 1, LIP UMR5668, France
edouard.bonnet@ens-lyon.fr

Nicolas Grelier

Department of Computer Science, ETH Zürich, Switzerland
nicolas.grelier@inf.ethz.ch

Tillmann Miltzow

Department of Information and Computing Sciences, Utrecht University, The Netherlands
t.miltzow@googlemail.com

Abstract

We study the complexity of Maximum Clique in intersection graphs of convex objects in the plane. On the algorithmic side, we extend the polynomial-time algorithm for unit disks [Clark '90, Raghavan and Spinrad '03] to translates of any fixed convex set. We also generalize the efficient polynomial-time approximation scheme (EPTAS) and subexponential algorithm for disks [Bonnet et al. '18, Bonamy et al. '18] to homothets of a fixed centrally symmetric convex set.

The main open question on that topic is the complexity of Maximum Clique in disk graphs. It is not known whether this problem is NP-hard. We observe that, so far, all the hardness proofs for Maximum Clique in intersection graph classes $\mathcal{I}$ follow the same road. They show that, for every graph G of a large-enough class $\mathcal{C}$, the complement of an even subdivision of G belongs to the intersection class $\mathcal{I}$. Then they conclude by invoking the hardness of Maximum Independent Set on the class $\mathcal{C}$, and the fact that the even subdivision preserves that hardness. However there is a strong evidence that this approach cannot work for disk graphs [Bonnet et al. '18]. We suggest a new approach, based on a problem that we dub Max Interval Permutation Avoidance, which we prove unlikely to have a subexponential-time approximation scheme. We transfer that hardness to Maximum Clique in intersection graphs of objects which can be either half-planes (or unit disks) or axis-parallel rectangles. That problem is not amenable to the previous approach. We hope that a scaled down (merely NP-hard) variant of Max Interval Permutation Avoidance could help making progress on the disk case, for instance by showing the NP-hardness for (convex) pseudo-disks.

2012 ACM Subject Classification Theory of computation → Computational geometry; Mathematics of computing → Graph algorithms

Keywords and phrases Disk Graphs, Intersection Graphs, Maximum Clique, Algorithms, NP-hardness, APX-hardness

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2020.17

Related Version A full version of the paper is available at `https://arxiv.org/abs/2003.02583`.

Funding *Nicolas Grelier*: Research supported by the Swiss National Science Foundation within the collaborative DACH project Arrangements and Drawings as SNSF Project 200021E-171681.
Tillmann Miltzow: NWO Veni Eager 016.Veni.192.250

1 Introduction

In an *intersection graph*, the vertices are represented by sets and there is an edge between two sets whenever they intersect. Of course if the sets are not restricted, every graph is an intersection graph. Interesting proper classes of intersection graphs are obtained by restricting the sets to be some specific geometric objects. This comprises unit interval, interval, multiple-interval, chordal, unit disk, disk, axis-parallel rectangle, segment, and string graphs, to name a few. For the most part, they transparently consist of all the intersection graphs of the corresponding objects. Note that *strings* are (polygonal) curves in the plane, and that chordal graphs are the intersection graphs of subtrees in a tree. Intersection graphs have given rise to books (see for instance [39], where applications to biology, psychology, and statistics, are detailed) chapters in monographs (as in [11]), surveys [24, 29], and theses [50]. In this paper we consider objects that are convex sets in the plane.

The Maximum Clique problem on geometric intersection graphs is especially interesting for two reasons. The first reason is that the gap between algorithmic upper and lower bounds is very large, when compared to other usual algorithmic questions on geometric intersection graphs. The second reason is that it has a very natural geometric interpretation, as we will explain below.

We start with a comparison of Maximum Clique with other standard algorithmic problems. The probably most studied problems on geometric intersection graphs are packing and covering problems, for which our theoretical understanding is rather comprehensive. Packing problems (such as Maximum Independent Set) and covering problems (such as Dominating Set) are often NP-hard in geometric intersection graphs since these problems are already hard on planar graphs. Note for instance that disk intersection graphs [34] and segment intersection graphs [17] both contain all the planar graphs. It turns out that Maximum Independent Set (MIS) and Dominating Set remain intractable in unit disk, unit square, or segment intersection graphs [37]: Not only are they NP-hard but, being W[1]-hard, they are unlikely to admit a fixed-parameter tractable (FPT), that is, $f(k)n^{O(1)}$-time, algorithm, with n being the input size, k the size of the solution, and f any computable function. This intractability is sharply complemented by PTASes for many problems [18, 44, 43, 23, 49, 50], whereas efficient PTASes (EPTASes) are ruled out by the W[1]-hardness of Marx [37]. The existence or unlikelihood of subexponential algorithms for various problems on segment and string graphs was investigated in [10].

On the contrary, many questions are still open when it comes to the computational complexity of Maximum Clique in intersection graphs. Clark et al. [19] show a polynomial-time algorithm for unit disks. A randomized EPTAS, deterministic PTAS, and subexponential-time algorithm were recently published, in top-level conferences [7, 6], for general disk graphs. However neither a polytime algorithm nor NP-hardness is currently known for Maximum Clique on disk graphs. Making progress on this open question is the main motivation of the paper. Maximum Clique was shown NP-hard in segment intersection graphs by Cabello et al. [15]. The proof actually carries over to intersection graphs of unit segments or rays. The existence of an FPT algorithm or of a subexponential-time algorithm for Maximum Clique in segment graphs are both open. Maximum Clique can be solved in polynomial-time in axis-parallel rectangle intersection graphs, since their number of maximal cliques is at most quadratic (every maximal clique corresponds to a distinct cell in any representation). This result was generalized to d-dimensional polytopes whose facets are all parallel to k fixed $(d-1)$-dimensional hyperplanes, where Maximum Clique can be solved in time $n^{O(dk^{d+1})}$ [13]. Note that if the rectangles may have arbitrary slopes, then Maximum Clique is NP-hard since the class then contains segment graphs.

The second reason to study Maximum Clique is that it translates into a very natural question: what is the maximum subset of pairwise intersecting objects? For unit disks, this is equivalent to looking for the maximum subset of centers with (geometric) diameter 2. This is a useful primitive in the context of clustering a given set of points. A related question with a long history is the number of points necessary and sufficient to pierce a collection of pairwise intersecting disks. Danzer [20] and Stacho [48] independently showed that four points are sufficient and sometimes necessary. Recently Har-Peled et al. [27] gave a linear-time algorithm to find five points piercing a pairwise intersecting collection of disks. A bit later, Carmi et al. [16] obtained a linear-time algorithm for only four points. Note that this implies that we can restrict our attention to instances that can be pierced by 4 points, both for polynomial-time algorithms and hardness reductions.

In this paper, our main focus is the complexity status of Maximum Clique on disk graphs. This is a long-standing and seemingly difficult open question. As we will detail in the next paragraphs, there has been basically only one approach to show NP-hardness of Maximum Clique in geometric intersection graph classes. This approach is somewhat doomed for the particular case of disk graphs. We develop a new way of showing conditional lower bounds in geometric intersection graphs. We believe that our approach faces a smaller barrier on its way to show that Maximum Clique is NP-hard on disk graphs.

A new alternative to the *co-2-subdivision* approach

Maximum Independent Set (MIS), which boils down to Maximum Clique on the complement graphs, is APX-hard on subcubic graphs [3]. A folklore self-reduction first discovered by Poljak [45] consists of subdividing each edge of the input graph twice (or any even number of times). One can show that this reduction preserves the APX-hardness. Therefore, a way to establish such an intractability for Maximum Clique on a given intersection graph class is to show that for every (subcubic) graph G, the complement of its 2-subdivision $\overline{\text{Subd}_2(G)}$ (or $\overline{\text{Subd}_s(G)}$ for a larger even integer s, see [25]) is representable. MIS admits a PTAS on planar graphs, but remains NP-hard. Hence showing that for every (subcubic) planar graph G, the complement of an even subdivision of G is representable shows the simple NP-hardness (see [15, 25]).

So far, representing complements of even subdivisions of graphs belonging to a class on which MIS is NP-hard (resp. APX-hard) has been the main, if not unique[1], approach to show the NP-hardness (resp. APX-hardness) of Maximum Clique in geometric intersection graph classes. This approach was used by Middendorf and Pfeiffer [40] for some restriction of string graphs, the so-called B_1-*VPG graphs*, by Cabello et al. [15] to settle the then long-standing open question of the complexity of Maximum Clique for segments (with the class of planar graphs), by Francis et al. [25] for 2-interval, unit 3-interval, 3-track, and unit 4-track graphs (with the class of all graphs; showing APX-hardness), and unit 2-interval and unit 3-track graphs (with the class of planar graphs; showing only NP-hardness), by Bonnet et al. [7] for filled ellipses and filled triangles, and by Bonamy et al. [6] for ball graphs, and 4-dimensional unit ball graphs.

One could hope that the same approach would carry over to show NP-hardness for disk intersection graphs. Bonnet et al. [7] give a structural insight that makes this idea unlikely to work. They showed that the complement of two mutually induced odd cycles is not a disk graph. As a consequence, to show the NP-hardness of Maximum Clique on disk graphs with the described approach, one can only hope to represent all the graphs without two mutually induced odd cycles. However we do not know if MIS is even NP-hard in that class. Classifying MIS as NP-hard or polynomial-time solvable on graph classes defined by a list of forbidden induced subgraphs has a long history. Yet the case when all pairs of mutually induced odd cycles are forbidden remains open. Specifically, showing NP-hardness for Maximum Clique on disk intersection graphs would resolve this graph-theoretic problem.

[1] Admittedly Butman et al. [14] showed that Maximum Clique is NP-hard on 3-interval graphs, by reducing from Max 2-DNF-SAT which is very close to Max Cut. However this result was later subsumed by [25].

The main conceptual contribution of the paper is to suggest an alternative to the standard approach. We introduce an intermediate problem that we call Max Interval Permutation Avoidance (MIPA, for short), which is a convenient way of seeing Max Cut. While the definition of MIPA may appear a bit technical (see Section 3), it will give a particularly fitting starting point to design transparent reductions to Maximum Clique in intersection graph classes (as exemplified by Theorem 8).

We prove that MIPA is NP-hard. We then transfer that lower bound to Maximum Clique in the intersection graphs of objects that can be either unit disks or axis-parallel rectangles; a class for which the *co-2-subdivision* approach does not seem to work. Recall that when all the objects are unit disks or when all the objects are axis-parallel rectangles, polynomial-time algorithms are known. The conceptual take-home message is that MIPA can give rise to new geometric hardness results when the *co-2-subdivision* approach fails. For our approach to be applicable to disks as well, it is important that it does not imply APX-hardness, as there is an EPTAS for Maximum Clique on disk graphs [7, 6]. As we even prove that MIPA is APX-hard, there are two options to tackle disk intersection graphs: Either design a reduction which specifically does *not* preserve the inapproximability gap, or restrict MIPA to a simpler NP-hard problem, one admitting a PTAS.

The latter idea of scaling MIPA down can, for instance, be done by replacing the arbitrary matching M by *pseudo-disk-like* objects, to a NP-hard problem admitting a PTAS. In doing so, one should keep in mind that Planar Max Cut [22] and Planar Not-All-Equal SAT [41] are solvable in polynomial-time. A next step could be to show the NP-hardness of Maximum Clique for (convex) pseudo-disks. It turns out to be already quite delicate. There is a distinct possibility that convex pseudo-disks have constant induced odd cycle packing number (see Section 2 for the definitions). This would imply a subexponential-time algorithm and an EPTAS [7, 6], and that one would need a scaled down version of MIPA to establish NP-hardness even in that case.

(a) Co-2-subdivision of subcubic graphs: edges are represented by an antimatching, vertices, by a clique.

(b) MIPA approach: vertices are represented by an antimatching with constant weight, edges, by a clique.

Figure 1 Dashed segments represent non-edges. Both the *co-2-subdivision* and the MIPA approaches require constructing an antimatching and a clique. In the *co-2-subdivision* approach, the *clique vertices* have co-degree 3 to the antimatching. In the MIPA approach their co-degree is only 2. While the difference is seemingly small, the graph class formed by axis-parallel rectangles and unit disks is not amenable to the *co-2-subdivision* approach (see Section 3).

In summary, our main contribution is a new distinct approach to show hardness of Maximum Clique in geometric intersection graphs. Although MIPA is only formally defined in Section 3, one can already see on Figure 1 that both approaches require representing an antimatching (i.e., a complement of an induced matching), a clique, and some relation between them. Antimatchings (and obviously cliques) of arbitrary size are representable by half-planes and unit disks. The difficulty in both cases is to get the right adjacencies between the antimatching and the clique. The MIPA approach only needs the vertices of the clique to *avoid* two vertices in the antimatching, whereas this number is at least three in the *co-2-subdivision* approach. This seemingly small difference is actually crucial, as we will see in Section 3.

Robustness

Up to this point, we remained vague on how the input intersection graph was given. For, say, disk graphs, do we receive the mere abstract graph or a list of the disks specified by their centers and radii? Computing the graph from the geometric representation can be done efficiently, but not the other way around. Indeed recognizing disk graphs is NP-hard [12] and even $\exists\mathbb{R}$-complete [32], where $\exists\mathbb{R}$ is a class between NP and PSPACE of all the problems polytime reducible to solving polynomial inequalities over the reals. Recognizing string graphs is NP-hard [35], and rather unexpectedly in NP [47], while recognizing segment graphs is $\exists\mathbb{R}$-complete [36]. In this context, an algorithm is said to be *robust* if it does not require the geometric representation. A polytime robust algorithm usually decides the problem for a *proper superclass* of the intersection graph class at hand, or correctly reports that the input does not belong to the class. Hence the robust algorithm does not imply an efficient recognition of the class. The polynomial-time algorithm of Clark et al. [19] for Maximum Clique in unit disk intersection graphs requires the geometric representation. Raghavan and Spinrad later extended it to an efficient robust algorithm [46].

Organization of the paper

The rest of the paper is organized as follows. In Section 2, we introduce the relevant set, graph, and geometry notations and definitions. Then we give the necessary background in hardness of approximation to get ready for the next section. In Section 3, we introduce the Max Interval Permutation Avoidance problem and prove that it is unlikely to admit a subexponential-time approximation scheme. We use it to show that adding half-planes or unit disks to axis-parallel rectangles is enough for Maximum Clique to go from trivially in P to APX-hard. This is a proof of concept for a different road-map to the *co-even-subdivision* approach, which is compromised for disk graphs. We also observe that if the half-planes are not allowed to be parallel (hence pairwise intersect), then the problem becomes tractable. In Section 4, we extend the EPTAS for disks [7, 6] to homothets of a fixed convex centrally symmetric set. In Section 5, we extend the polytime algorithm for unit disks [19, 46] to translates of a fixed convex set. Our algorithms are robust and our lower bound also holds when the geometric representation is given. Due to lack of space, theorems and lemmas marked with the $\star$ symbol have their proof deferred to the full version of the paper [8]. Some results had to be moved to the full version entirely.

2 Preliminaries

Sets and graphs

For a pair of positive integers $i \leqslant j$, $[i, j]$ denotes the set of all the integers that are at least i and at most j, and $[i]$ is a short-hand for $[1, i]$. We overload the notation $[\cdot, \cdot]$: If it is explicit or clear from the context that x or y is non-integral, then $[x, y]$ denotes the set of reals that are at least x and at most y. We use the usual notations and definitions of graph theory, as they can be found for example in Diestel's book [21]. We denote by K_t, C_t, and $K_{s,t}$ the complete graph (or clique) on t vertices, the cycle on t vertices, and the biclique on s and t vertices. The graph $\overline{G}$ denotes the *complement* of G, obtained by flipping edges into non-edges, and non-edges into edges. Subdividing an edge $e = uv$ consists of adding a new vertex linked to both u and v, and removing the edge e. The 2-subdivision $\text{Subd}_2(G)$ of a graph G is obtained by subdividing each of its edges twice, hence replacing them by paths of three edges. An even subdivision of a graph G consists of subdividing every edge of G

an even number of times (potentially zero). A cycle is said to be *induced* if it is chordless. An *odd cycle* (resp. *even cycle*) is a cycle on an odd (resp. even) number of vertices. One can observe that an odd cycle always contains an induced odd cycle. Two cycles are said *mutually induced* if they are chordless and there is no edge linking a vertex of one to a vertex of the other. The induced odd cycle packing number is the maximum number of disjoint odd cycles, that are pairwise mutually induced. An *antimatching* is the complement of an *induced matching* (i.e., a disjoint union of edges). We say that a graph G is *representable* by some geometric objects, if translates of these objects may have G as intersection graph.

Geometric notations

In this paper, we only consider sets in the plane. For two distinct points a and b, $\ell(a, b)$ denotes the line going through a and b. A set S is *convex* if for any two distinct points a and b in S, the line segment with endpoints a and b is contained in S. It is *bounded*, if it is contained in some disk. A set S is said to be *centrally symmetric* about the origin if for any point a in S, $-a$ is also in S. We mostly deal with sets that are bounded, centrally symmetric, and convex, as they are a natural generalization of disks.

For two sets S_1 and S_2, we denote by $S_1 + S_2 := \{s_1 + s_2 \mid s_1 \in S_1, s_2 \in S_2\}$ their Minkowski sum. For the sake of simplicity, for any point c and any set S, we denote by $c + S$ the Minkowski sum of $\{c\}$ and S. S' is a *translate* of S if there exists c such that $S' = c + S$. Given a positive real number λ, λS denotes the set $\{\lambda s \mid s \in S\}$. We say that S' is a *homothet* of S if there exist a positive real number λ and a point c such that $S' = c + \lambda S$. Moreover we name c the *center* of S', and λ its *scaling factor*.

Let F be a family of sets in the plane. They form a *pseudo-disk arrangement* if for any pair of sets of F, their boundaries intersect at most twice. If the sets are also convex we refer to them as *convex pseudo-disks*. They also constitute a natural generalization of disks. A set of rectangles are *axis-parallel* if their boundaries all share the same two slopes. A rectangle is an ε-square if its length divided by its width is smaller than $1 + \varepsilon$.

Approximation-schemes

A *polynomial-time approximation scheme* (PTAS) for a maximization problem is an algorithm which takes, together with its input, a parameter $\varepsilon > 0$ and outputs in time $n^{f(\varepsilon)}$ a solution of value at least $(1 - \varepsilon)$OPT, where OPT is the optimum value. An *efficient* PTAS (EPTAS) is the same but has running time $f(\varepsilon)n^{O(1)}$. Note that the existence of an EPTAS, for a problem in which the objective value is the size of the solution k, implies an FPT algorithm in k, by setting ε to $1 - \frac{1}{k+1}$. Indeed in time $f(1 - \frac{1}{k+1})n^{O(1)} = g(k)n^{O(1)}$, one then obtains an *exact* solution. A *quasi* PTAS (QPTAS) is an approximation scheme with running time $n^{\text{polylog}\, n}$, for every $\varepsilon > 0$. Less standardly, we call *subexponential* AS (SUBEXPAS) an approximation scheme with running time 2^{n^γ} for some $\gamma < 1$, for every $\varepsilon > 0$. These approximation schemes can come deterministic or randomized. A maximization problem Π is *APX-hard* if there is a constant $\gamma < 1$ such that γ-approximating Π is NP-hard. Unless P=NP, an APX-hard problem cannot admit a PTAS. Ruling out a SUBEXPAS (under admittedly a stronger assumption than P≠NP) constitutes a sharper inapproximability than the APX-hardness.

Strong inapproximability of Positive Not-All-Equal 3-SAT-3. A longer version of this subsection can be found in the full version of this paper [8]. The Exponential-Time Hypothesis (ETH, for short) of Impagliazzo and Paturi [30] asserts that there is an $s_3 > 0$ (taking the same notation as in the original paper) such that 3-SAT cannot be solved in time $2^{s_3 n}$ on n-variable instances. By the Sparsification Lemma [31], the ETH implies the same lower bound for 3-SAT-B, in which every variable appears at most a constant B number of times, depending only on s_3. Our starting point combines some sharp polytime inapproximability [28], a PCP construction [42], and the Sparsification Lemma.

▶ **Theorem 1.** *[28, 42, 31] Under the ETH, for every $\delta > 0$ one cannot distinguish in time $2^{n^{1-\delta}}$, n-variable m-clause 3-SAT-instances that are satisfiable from instances where at most $(7/8 + o(1))m$ clauses can be satisfied, even when each variable appears in at most B clauses. Thus 3-SAT-B cannot be $7/8 + o(1)$-approximated in time $2^{n^{1-\delta}}$.*

We recall the definition of Not-All-Equal k-SAT (NAE k-SAT, for short).

Not-All-Equal k-SAT
Input: A conjunction of m "clauses" $\phi = \bigwedge_{i \in [m]} C_i$ each on at most k literals.
Goal: Find a truth assignment of the n variables such that each "clause" has at least one satisfied literal and at least one non-satisfied literal.

The Not-All-Equal k-SAT-B-problem is the same but each variable appears in at most B clauses (similarly as for k-SAT-B). The adjective Positive prepended to a satisfiability problem means that no negation (or *negative literal*) can appear in its instances.

▶ **Theorem 2.** [$\star$] *Under the ETH, for every $\delta > 0$ one cannot distinguish in time $2^{n^{1-\delta}}$, n-variable m-clause Not-All-Equal 4-SAT-instances that are satisfiable from instances where at most $4991m/5000$ clauses can be satisfied, even when each variable appears in at most B clauses. Thus Not-All-Equal 4-SAT-B cannot be $4991/5000$-approximated in time $2^{n^{1-\delta}}$.*

We now decrease the size of the clauses to at most 3. The next reduction and the subsequent one are folklore. We give complete proofs both for the sake of self-containment and to report explicit inapproximability bounds.

▶ **Theorem 3.** [$\star$] *Under the ETH, for every $\delta > 0$ one cannot distinguish in time $2^{n^{1-\delta}}$, n-variable m-clause Not-All-Equal 3-SAT-instances that are satisfiable from instances where at most $9991m/10000$ clauses can be satisfied, even when each variable appears in at most B clauses. Thus Not-All-Equal 3-SAT-B cannot be $9991/10000$-approximated in time $2^{n^{1-\delta}}$.*

Finally, by a linear reduction from Not-All-Equal 3-SAT-B to Positive Not-All-Equal 3-SAT-3, we decrease the maximum number of occurrences per variable to 3, and we remove the negative literals. A compact yet weaker implication of the following theorem is that a QPTAS for Positive Not-All-Equal 3-SAT-3 would disprove the ETH.

▶ **Theorem 4.** [$\star$] *Under the ETH, for every $\delta > 0$ one cannot distinguish in time $2^{n^{1-\delta}}$, n-variable m-clause Positive Not-All-Equal 3-SAT-3-instances that are satisfiable from instances where at most γm clauses can be satisfied, with $\gamma := (60000B^2 - 9)/(60000B^2)$. Thus Positive Not-All-Equal 3-SAT-3 cannot be γ-approximated in time $2^{n^{1-\delta}}$.*

This last reduction no longer implies APX-hardness. Indeed, the value B in the inapproximability ratio is finite only if $s_3 > 0$. So one should assume the ETH, and not the mere P $\neq$ NP, to rule out an approximation algorithm with ratio $\gamma < 1$. Sacrificing the strong lower bound in the running time, we can overcome that issue. Berman and Karpinski [5] showed that it is NP-hard to approximate Max 2-SAT-3 within ratio better than $787/788$. Following the reduction of Theorem 2 from Max 2-SAT-3, we derive the following inapproximability.

▶ **Corollary 5.** *Approximating NAE 3-SAT-10 within ratio* 51326/51327 *is NP-hard.*

Proof. Observe that the clause size grows from 2 to 3, and that the variables z_j are part of at most 9 clauses. ◀

Then following Theorem 4, we get:

▶ **Corollary 6.** *Approximating* POSITIVE *NAE 3-SAT-3 within ratio* 49888956/49888957 *is NP-hard.*

3 Max Interval Permutation Avoidance, unit disks and rectangles

We introduce the MAX INTERVAL PERMUTATION AVOIDANCE-problem (MIPA, for short), a convenient intermediate problem to show APX-hardness for geometric problems. We start with an informal description. Let M be a perfect matching between the n points $[n] \times \{0\}$ and $[n] \times \{1\}$, in $\mathbb{N}^2$. This matching can be represented by a permutation σ, such that for every $i \in [n]$, $(i, 0)$ is matched with $(\sigma(i), 1)$. Imagine now a set of intervals on the line $y = 1/2$ whose endpoints are all in $[n]$. The aim is to move each interval "up" or "down", by translating it by $(0, 1/2)$ or by $(0, -1/2)$, respectively, such that the number of edges of M with no endpoint on a translated interval is maximized. An edge of M with at least one endpoint in a *moved* (or *positioned*) interval is said to be *covered* or *destroyed*. The edge is said to be *uncovered* or *preserved* otherwise. Equivalently MAX INTERVAL PERMUTATION AVOIDANCE aims to minimize the number of covered edges, or maximize the number of uncovered edges. We choose the maximization formulation, since we will both reduce from a maximization problem (POSITIVE NOT-ALL-EQUAL 3-SAT-3) and to a maximization problem (MAXIMUM CLIQUE on disks and rectangles). Thus the objective value will be the number of *uncovered edges*.

MAX INTERVAL PERMUTATION AVOIDANCE
Input: A permutation σ over $[n]$ representing a perfect matching M between the points $(1, 0), (2, 0), \ldots, (n, 0)$ and $(\sigma(1), 1), (\sigma(2), 1), \ldots, (\sigma(n), 1)$ respectively, and a set of integer ranges $\mathcal{I} := \{I_1, \ldots, I_h\}$ where $I_k := [\ell_k, r_k]$ and $1 \leqslant \ell_k \leqslant r_k \leqslant n$, for every $k \in [h]$.
Goal: A placement function $p : \mathcal{I} \to \{0, 1\}$ encoding that interval I_k has its endpoints positioned in $(\ell_k, p(I_k))$ and $(r_k, p(I_k))$, which maximizes the number of edges of M with no endpoint on a positioned interval.

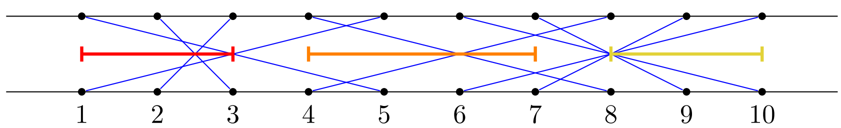

Figure 2 An example of a symmetric instance of MIPA with three disjoint ranges. An optimum solution puts the second interval opposite to the first and third intervals. This leaves 4 edges of the matching uncovered.

A MIPA-input may interchangeably be given as $(\sigma, \mathcal{I})$ or as $(M, \mathcal{I})$. One may observe that a *constant* placement (i.e., $p(I_1) = \ldots = p(I_h) = 0$, or $p(I_1) = \ldots = p(I_h) = 1$) is a worse solution when the intervals of $\mathcal{I}$ span $[n]$, since it covers all the edges of M. We say that the matching M is *symmetric* if $(i, 0)(j, 1) \in M$ implies that $(i, 1)(j, 0) \in M$, for every

$i, j \in [n]$; in the geometric viewpoint, it is equivalent to $y = 1/2$ being a symmetry axis of M, and in the permutation viewpoint, it is equivalent to σ being a product of pairwise-disjoint transpositions. Other handy (as far as hardness of geometric problems is concerned) technical problems involving intervals and/or permutations include Crossing-Avoiding Matching in Guśpiel [26] or Crossing-Minimizing Perfect Matching in Guśpiel et al. [2], the problem of covering a 2-track point set by selecting k 2-track intervals [38] or Structured 2-Track Hitting Set [9]. It is no coincidence that these convenient starting problems all involve matchings/permutations and/or intervals. Indeed the latter objects are more easily encoded in a geometric setting than their generalizations: arbitrary binary relations and arbitrary sets. Later we will see how disks can encode intervals and how rectangles can encode a permutation, in the context of the Maximum Clique-problem.

We rule out an approximation scheme for Max Interval Permutation Avoidance, even if subexponential-time is allowed. In particular a QPTAS for MIPA is highly unlikely. We recall that $\gamma = (60000B^2 - 9)/(60000B^2)$ and that B is a finite integral constant, assuming the ETH ($s_3 > 0$).

▶ **Lemma 7.** *For every $\delta > 0$, Max Interval Permutation Avoidance cannot be γ'-approximated in time $2^{|M|^{1-\delta}}$, with $\gamma' := 1 - (1-\gamma)/13 < 1$, unless the ETH fails. Furthermore, Max Interval Permutation Avoidance is NP-hard and APX-hard. These results hold even if the length of every interval of $\mathcal{I}$ is at most 5, and the matching M is symmetric.*

Proof. We give a reduction ϕ from Positive Not-All-Equal 3-SAT-3 to Max Interval Permutation Avoidance. Let ϕ be a Positive NAE 3-SAT-3-instance, with variables $x_1, \ldots, x_n$ and clause $C_1, \ldots, C_m$. For every $x_i \in C_j$, we denote by $\text{occ}(x_i, C_j)$ the number of occurrences of x_i in the clauses $C_1, \ldots, C_j$. We observe that $\text{occ}(x_i, C_j) \in \{1, 2, 3\}$. We build an instance $\rho(\phi) := (M, \mathcal{I})$ of MIPA in the following way. For each variable x_i of ϕ, we reserve a range $[3(i-1)+1, 3(i-1)+3]$ with 3 integral points on both lines $y = 0$ and $y = 1$. These points will be matched by M to points in the clause gadgets. We add the interval $X_i := [3(i-1)+1, 3(i-1)+3]$ to $\mathcal{I}$. We now describe the 2-clause and the 3-clause gadgets.

For every 2-clause $C_j := x_a \lor x_b$, we allocate a slot S_j of size 10 (on $y = 0$ and $y = 1$) appended to the current last position. The first half of S_j, that is, the indices in $[s_j, s_j+4]$ of S_j correspond to x_a, while the indices in $[s_j+5, s_j+9]$ correspond to x_b. For every $(d_1, d_2) \in \{(0,1), (1,0)\}$ and $h \in [4]$, we add to M the edge between (s_j+h, d_1) and (s_j+5+h, d_2). We add the intervals $C_j(x_a) := [s_j, s_j+4]$ and $C_j(x_b) := [s_j+5, s_j+9]$ to $\mathcal{I}$. Finally for each $(d_1, d_2) \in \{(0,1), (1,0)\}$, we add to M the edges between (s_j, d_1) and $(3(a-1) + \text{occ}(x_a, C_j), d_2)$, and between (s_j+5, d_1) and $(3(b-1) + \text{occ}(x_b, C_j), d_2)$.

For every 3-clause $C_j := x_a \lor x_b \lor x_c$, we allocate a slot S_j of size 15 (on $y = 0$ and $y = 1$) appended to the current last position. The first third of S_j, that is, the indices in $[s_j, s_j+4]$ of S_j correspond to x_a, the second third, the indices in $[s_j+5, s_j+9]$ correspond to x_b, and the last third, the indices in $[s_j+10, s_j+14]$ correspond to x_c. We add the intervals $C_j(x_a) := [s_j, s_j+4]$, $C_j(x_b) := [s_j+5, s_j+9]$, and $C_j(x_c) := [s_j+10, s_j+14]$ to $\mathcal{I}$. Similarly for every $(d_1, d_2) \in \{(0,1), (1,0)\}$ and $(h, p) \in \{(a, 0), (b, 1), (c, 2)\}$, we add to M the edge between (s_j+5p, d_1) and $(3(h-1) + \text{occ}(x_h, C_j), d_2)$. We call these edges *internal* (same for the 2-clause gadget). Finally we add to M four edges from every pair of ranges in $\{[s_j, s_j+4], [s_j+5, s_j+9], [s_j+10, s_j+14]\}$, two starting on the line $y = 0$ (ending on $y = 1$) and two starting on $y = 1$ (ending on $y = 0$). We call these edges *variable-clause* (same for the 2-clause gadget).

For each variable x_i with only two occurrences in ϕ, we link its third occurrence pair to a dummy pair $(d_i, 0), (d_i, 1)$, appended to the current last position. That is, we add the edges $(3(i-1)+3, 0)(d_i, 1)$ and $(3(i-1)+3, 1)(d_i, 0)$ to M. Although not needed, we also add the singleton interval $D_i := \{d_i\}$ to $\mathcal{I}$. We call it *dummy gadget* and consider it as a special case of a clause gadget. This finishes the construction of the MIPA-instance $(M, \mathcal{I})$. Observe that every point is matched, and that all the intervals of $\mathcal{I}$ are pairwise disjoint, and of length at most 5. The perfect matching M comprises at most $3n + 15m + n \leqslant 49n$ edges.

We assume that ϕ is satisfiable, and let $\mathcal{V}$ be a satisfying assignment. We build the following solution to the MIPA-instance. We push the interval X_i up (to the line $y = 1$) if x_i is set to true by $\mathcal{V}$, and we push it down (to the line $y = 0$) otherwise. In the clause gadgets (and dummy gadgets), we do the opposite: we push $C_j(x_i)$ (D_i) down if x_i is set to true, and up if x_i is set to false. This solution preserves four edges within each clause gadget, and an additional $3n$ edges between the variable gadgets and the clause gadgets. Hence the total number of preserved edges is $4m + 3n$.

We now assume that at most γm clauses of ϕ are satisfiable. Let p be a placement function of the intervals of $\mathcal{I}$, maximizing the number of preserved edges of M. We first argue that not giving the same placement (up/1 or down/0) to the three (resp. two) intervals $C_j(x_a), C_j(x_b), C_j(x_c)$ (resp. $C_j(x_a), C_j(x_b)$) of a 3-clause gadget (resp. 2-clause gadget) is always better. Note that any equal placement destroys all the edges of M internal to the clause gadget of C_j, and preserves at most three variable-clause edges. On the other hand, a placement with at least one interval on each side preserves already four internal edges. We can then assume that p does not give equal placement in any clause gadget. Let $\mathcal{V}$ be the assignment of the variables of ϕ which sets x_i to true if $p(X_i) = 1$, and to false, if $p(X_i) = 0$. By assumption $\mathcal{V}$ does not satisfy at least $(1-\gamma)m$ clauses. In each corresponding clause gadget, one can preserve at most two variable-clause edges of M. Indeed, since ϕ is a Positive NAE 3-SAT-3-instance, all three variable-clause edges incident to the clause gadget and not covered by the placement of the X_i land on the same side. By the previous remark, at least one such edge should be destroyed (to preserve four internal edges). Thus the placement p preserves at most $3n + 4m - (1-\gamma)m$ edges.

Since $|M| = O(n+m) = O(n)$ and $\frac{3n+4m-(1-\gamma)m}{3n+4m} \leqslant 1 - \frac{1-\gamma}{13}$, by Theorem 4 MIPA cannot be γ'-approximated in time $2^{|M|^{1-\delta}}$, under the ETH. Besides, by Corollary 6, MIPA cannot be 648556435/648556436-approximated in polynomial-time, unless P=NP. In particular, this problem is NP-hard and even APX-hard. ◀

We recall that Maximum Clique can be solved in polynomial-time in unit disk graphs [19, 46] and in axis-parallel rectangle intersection graphs [13]. Now if the objects can be unit disks *and* axis-parallel rectangles, we show that even a SUBEXPAS is unlikely. We denote by {Obj,Obj'}-Maximum Clique the clique problem in the intersection graphs of objects that can be either Obj or Obj'.

▶ **Theorem 8.** *For every $\delta > 0$, Maximum Clique in intersection graphs G of unit disks and axis-parallel rectangles cannot be c-approximated in time $2^{|V(G)|^{1-\delta}}$, with $c := 1 - (1-\gamma)/153 < 1$, unless the ETH fails. Moreover, this problem is NP-hard and APX-hard.*

Proof. We give a reduction from Max Interval Permutation Avoidance to {Unit Disks, Axis-Parallel Rectangles}-Maximum Clique or {Half-Planes, Axis-Parallel Rectangles}-Maximum Clique. Let $(M, \mathcal{I})$ be an instance of MIPA over $[n]$, where M is symmetric, and all the intervals of $\mathcal{I}$ have size at most 5. We build the following set of axis-parallel rectangles $\mathcal{R}$ and half-planes $\mathcal{H}$. See Figure 3 for an illustration.

Let O be the origin of the plane. We place from left to right $n+2$ points $p_0, p_1, \ldots, p_n, p_{n+1}$ on a convex curve in the top-left quadrant, say $x \mapsto -1/x$ on $[-(1+\lambda), -1]$ for some small $\lambda > 0$. We wiggle the points p_i so that for every $i \leqslant j \in [n]$, the slope of the line passing through middle(p_{i-1}, p_i) and middle(p_j, p_{j+1}) has a distinct value, where middle(p, q) denotes the midpoint of the segment with endpoints p and q. We define $q_0, q_1, \ldots, q_n, q_{n+1}$, such that O is the middle of the segment $p_i q_i$ for every $i \in [0, n+1]$. In other words, this new chain is obtained by central symmetry about O. Observe that sorted by x-coordinates, these $2n+4$ points read $p_0, p_1, \ldots, p_n, p_{n+1}, q_{n+1}, q_n, \ldots, q_1, q_0$. The points $p_1, \ldots, p_n$ represent $[n] \times \{0\}$ in the MIPA-instance, while the points $q_1, \ldots, q_n$ represent $[n] \times \{1\}$.

For every pair $i \leqslant j \in [n]$, we can associate a line $\ell_p(i,j)$ passing through middle(p_{i-1}, p_i) and middle(p_j, p_{j+1}). Notice that, by convexity, $\ell_p(i,j)$ separates the points $p_i, p_{i+1}, \ldots, p_{j-1}, p_j$ (below it) from the points $p_1, \ldots, p_{i-1}, p_{j+1}, \ldots, p_n$ (above it). We similarly define $\ell_q(i,j)$ as the line passing through middle(q_{i-1}, q_i) and middle(q_j, q_{j+1}). We observe that $\ell_p(i,j)$ and $\ell_q(i,j)$ are parallel. For every interval $I = [i,j] \in \mathcal{I}$, we introduce in the Maximum Clique-instance the half-plane $h_p(I) := h_p(i,j)$ as the closed upper half-plane whose boundary is $\ell_p(i,j)$, and $h_q(I) := h_q(i,j)$ as the closed lower half-plane whose boundary is $\ell_q(i,j)$. We give these two objects weight 5 by superimposing 5 copies of them. All pairs of introduced half-planes intersect, except the pairs $\{h_p(i,j), h_q(i,j)\}$.

Finally for every edge $(i,0)(j,1)$ of the matching M (with $i,j \in [n]$), we add an axis-parallel rectangle $R(i,j)$ whose top-left corner is p_i and bottom-right corner is q_j. This finishes the construction of $(\mathcal{R}, \mathcal{H})$. When λ tends to 0, the rectangles are arbitrary close to squares of equal side-length. In other words, for any $\varepsilon > 0$, the axis-parallel rectangles can be made ϵ-squares. The half-planes can be turned into unit disks, making the side-length of the rectangles very small compared to 1. We denote by $(\mathcal{R}, \mathcal{D})$ the corresponding sets of axis-parallel rectangles and unit disks, and by G their intersection graph.

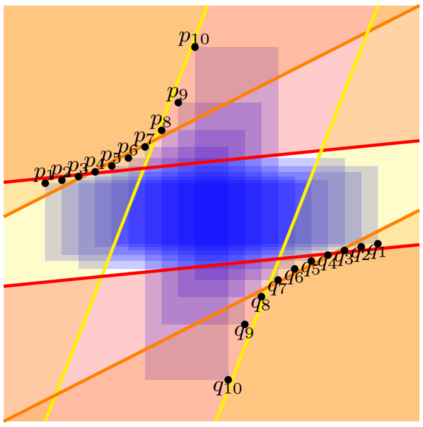

Figure 3 The output of the reduction on the instance of Figure 2.

Let us consider instances of MIPA produced by the previous reduction from Positive NAE 3-SAT-3, on ν-variable μ-clause formulas that are either satisfiable or with at least $(1-\gamma)\mu$ non satisfiable clauses. We call *yes-instances* the former MIPA-instances, and *no-instances*, the latter. If $(M, \mathcal{I})$ is a yes-instance, we claim that G has a clique of size $5|\mathcal{I}|+3\nu+4\mu$. Indeed there is a placement p that preserves $3\nu+4\mu$ edges of M. We start by taking in the clique all the half-planes (or corresponding unit disks) $h_p(I)$ whenever $p(I)=0$, and $h_q(I)$ whenever $p(I)=1$. Since these objects have weight 5 (actually 5 stacked copies), this amounts to $5|\mathcal{I}|$ vertices. The corresponding half-planes pairwise intersect since their boundaries have distinct slopes. Then we include to the clique the $3\nu+4\mu$ rectangles $R(i,j) \in \mathcal{R}$ such that $(i,0)(j,1)$ is preserved by p. All the rectangles pairwise intersect since they all contain the origin O. Every pair of chosen half-plane $h_z(I)$ $(z \in \{p,q\})$ and rectangle $R(a,b)$ intersects, otherwise the placement of I would cover $(a,0)(b,1)$. Thus we exhibited a clique of size $5|\mathcal{I}|+3\nu+4\mu$ in G.

We now assume that $(M, \mathcal{I})$ is a no-instance, and we claim that G has no clique larger than $5|\mathcal{I}|+3\nu + 4\mu - (1-\gamma)\mu$. Let us see how to build a clique in G. One can take at most one object between $h_p(I)$ and $h_q(I)$ (since they do not intersect). There is a maximum clique that takes at least one of $h_p(I)$ and $h_q(I)$ since $h_p(I)$ has weight 5 and intersects every object but $h_q(I)$ plus at most 5 rectangles (recall that the intervals of $\mathcal{I}$ have size at most 5). Thus we assume that our maximum clique takes exactly one object between $h_p(I)$ and $h_q(I)$, for every $I \in \mathcal{I}$. We consider the placement p defined as $p(I) = 0$ if $h_p(I)$ is in the clique, and $p(I) = 1$ if $h_q(I)$ is in the clique. Now the rectangles $R(i, j)$ that are adjacent to the chosen half-planes of $\mathcal{H}$ (or unit disks of $\mathcal{D}$) correspond to the edges $(i, 0)(j, 1)$ of M which are preserved. By Lemma 7, there are at most $3\nu + 4\mu - (1-\gamma)\mu$ such rectangles.

Since $|V(G)|= |\mathcal{H}|+|\mathcal{R}|= 10|\mathcal{I}|+|M|= O(\nu + \mu) = O(\nu)$ and $\frac{5|I|+3\nu+4\mu-(1-\gamma)\mu}{5|I|+3\nu+4\mu} \leqslant 1 - \frac{(1-\gamma)\mu}{140\mu+9\mu+4\mu} = 1 - \frac{1-\gamma}{153} = c$, by Theorem 4, {Half-Planes/Unit Disks, Axis-Parallel Rectangles}-Maximum Clique cannot be c-approximated in time $2^{|V(G)|^{1-\delta}}$, under the ETH. Additionally, by Corollary 6, this problem cannot be 7633010347/7633010348-approximated in polynomial-time, unless P=NP. In particular, it is NP-hard and even APX-hard. ◀

We observe that if all the half-planes pairwise intersect (for instance because their boundaries are assumed to have distinct slopes), then there is a polynomial-time algorithm, given a geometric representation. Let again $\mathcal{H}$ be the half-planes and $\mathcal{R}$, the axis-parallel rectangles, in the representation of the graph G. Recall that the number of maximal cliques in $G[\mathcal{R}]$ is polynomial, and that they can be enumerated efficiently. For each maximal clique $\mathcal{R}_c \subseteq \mathcal{R}$, we compute the maximum clique in the co-bipartite graph $G[\mathcal{H} \cup \mathcal{R}_c]$. This is thus equivalent to computing MIS in a bipartite graph. Due to Kőnig's theorem, this can be done in polynomial-time by a matching algorithm. We output C the largest clique that we find. C is a maximum clique in G, since $C \cap \mathcal{R}$ is by definition a clique, so it is contained in a maximal clique of $G[\mathcal{R}]$.

Let us briefly discuss the issue the *co-2-subdivision* approach encounters for {Half-Planes, Axis-Parallel Rectangles}-Maximum Clique. Axis-parallel rectangles cannot represent a large antimatching (they already cannot represent $\overline{3K_2}$). Hence, as in our construction, the large antimatching has to be, for the most part, realized by half-planes. Now in the MIPA approach, the axis-parallel rectangles can avoid *two* arbitrary half-planes with the freedom of their top-left and bottom-right corners. In the *co-2-subdivision* approach, they would have to avoid *at least three* arbitrary half-planes, and do not have enough degrees of freedom for that.

4 Homothets of a centrally symmetric convex set

Here we observe that the EPTAS for Maximum Clique in disk graphs extends to the intersection graphs of homothets of a centrally symmetric convex set. Bonamy et al. show:

▶ **Theorem 9** ([6]). *For any constants $d \in \mathbb{N}$, $0 < \beta \leqslant 1$, for every $0 < \varepsilon < 1$, there is a randomized $(1-\varepsilon)$-approximation algorithm running in time $2^{\tilde{O}(1/\varepsilon^3)}n^{O(1)}$, and a deterministic PTAS running in time $n^{\tilde{O}(1/\varepsilon^3)}$ for Maximum Clique on n-vertex graphs G satisfying the following conditions:*

- *there are no two mutually induced odd cycles in $\overline{G}$ (the complement of G),*
- *the VC-dimension of the neighborhood hypergraph $\{N[v] \mid v \in V(G)\}$ is at most d, and*
- *G has a clique of size at least βn.*

The first item is enough to obtain a subexponential-algorithm [7] and boils down to proving a structural lemma on the representation of $K_{2,2}$ (see Lemma 11). We show that the previous theorem applies to more general shapes than disks.

▶ **Theorem 10.** *Maximum Clique admits a subexponential-time algorithm and an EPTAS in intersection graphs of homothets of a fixed bounded centrally symmetric convex set S.*

Let S be a centrally symmetric, bounded, convex set. We can define a corresponding norm as follow: for any $x \in \mathbb{R}^2$, let $\|x\|$ be equal to $\inf\{\lambda > 0 \mid x \in \lambda S\}$. This is well-defined since S is bounded. It is absolutely homogeneous because S is centrally symmetric, and it is subadditive because S is convex. Therefore $\|.\|$ is a norm. We use the norm we have defined, and check the three conditions of Theorem 9.

▶ **Lemma 11.** *In a representation of $K_{2,2}$ with homothets of S placing the four centers in convex position, the non-edges are between vertices corresponding to opposite corners of the quadrangle.*

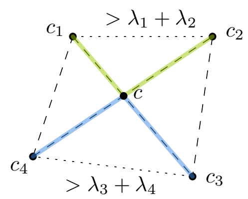

Figure 4 Illustration of the proof of Lemma 11. Non-edges are dotted and edges are dashed.

Proof. Let S_1, S_2, S_3 and S_4 be the four homothets. We denote by c_i the center of S_i, and by λ_i its scaling factor. Let us assume by contradiction that they appear in this order on the convex hull, that S_1 and S_2 make one non-edge, and that S_3 and S_4 make the other. By assumption, we have $\|c_1 - c_2\| > \lambda_1 + \lambda_2$, and likewise $\|c_3 - c_4\| > \lambda_3 + \lambda_4$. Let us denote by c the intersection of the lines $\ell(c_1, c_3)$ and $\ell(c_2, c_4)$, where $\ell(p, q)$ denotes the line going through two distinct points p and q. We have $\|c_1 - c\| + \|c - c_2\| > \|c_1 - c_2\|$ by triangular inequality. Likewise it holds $\|c_3 - c\| + \|c - c_4\| > \|c_3 - c_4\|$. We therefore obtain $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 < \|c_1 - c\| + \|c - c_2\| + \|c_3 - c\| + \|c - c_4\| = \|c_1 - c_3\| + \|c_2 - c_4\| \leqslant \lambda_1 + \lambda_3 + \lambda_2 + \lambda_4$, which is a contradiction. ◀

Lemma 11 implies by some parity arguments that the first condition of Theorem 9 holds (see Theorem 6 in [7]). It is well known that a family of homothets forms a pseudo-disk arrangement. Therefore the second property holds as shown by Aronov et al. [4]. Finally we enforce the third condition of Theorem 9, by using a chi-boundedness result of Kim et al. [33].

▶ **Lemma 12.** *With a polynomial multiplicative factor in the running time, one can reduce to instances satisfying the third condition of Theorem 9 with $\beta = 1/36$.*

Proof. Kim et al. [33] show that in any representation of an intersection graph G of homothets of a convex set, a homothet S with a smallest scaling factor has degree at most $6\omega(G) - 7$, where $\omega(G)$ denotes the clique number of G. Their proof also implies that the independence number of its neighborhood is at most 6. By degenerence, the coloring number, denoted by $\chi(G)$ is at most $6\omega(G) - 6$. First we find in polynomial-time a vertex v such that the independence number of its neighborhood is at most 6. Let us denote by G_v the subgraph induced by its neighborhood, and n denotes its number of vertices. We denote by $\alpha(.)$ the independence number of a graph. As G_v has a representation with homothets of S, we have $\chi(G_v) \leqslant 6\omega(G_v)$. Therefore $\alpha(G_v)\omega(G_v) \geqslant \frac{1}{6}\alpha(G_v)\chi(G_v) \geqslant \frac{1}{6}n$. Thus by assumption we have $\omega(G_v) \geqslant \frac{1}{36}n$. Then we can compute a maximum clique that contains v, or remove v from the graph and iterate. The EPTAS of Bonamy et al. is called linearly many times. ◀

5 Translates of a convex set

We show in this section that we can extend the algorithm of Clark et al. [19] and its robust version [46] from unit disks to any centrally symmetric, bounded, convex set.

▶ **Theorem 13.** *Maximum Clique admits a robust polynomial-time algorithm in intersection graphs of translates of a fixed centrally symmetric, bounded, convex set.*

Moreover, as shown by Aamand et al. [1], for every bounded and convex set S_1, there exists a centrally symmetric, bounded and convex set S_2 such that $\mathcal{G}_{S_1} = \mathcal{G}_{S_2}$, where $\mathcal{G}_S$ denotes the intersection graphs class of translates of S. Thus we obtain the immediate corollary:

▶ **Corollary 14.** *Maximum Clique admits a robust polynomial-time algorithm in intersection graphs of translates of a fixed bounded and convex set.*

We prove Theorem 13 in two steps. First we show how to compute in polynomial time a maximum clique when a representation is given. Secondly we use the result by Raghavan and Spinrad [46] to obtain a robust algorithm.

Let S be a centrally symmetric, bounded, convex set. We use the norm defined in Section 4: for any $x \in \mathbb{R}^2$, let $\|x\|$ be equal to $\inf\{\lambda > 0 \mid x \in \lambda S\}$. Let S_1 and S_2 be two translates of S, with respective centers c_1 and c_2. Remark that S_1 and S_2 intersect if and only if $\|c_1 - c_2\| \leqslant 2$. Let us assume that $d := \|c_1 - c_2\| \leqslant 2$. We denote by S' the set S scaled by d: $S' := dS$, and we then define: $D := \{x \in \mathbb{R}^2 \mid \|x - c_1\| \leqslant d, \|x - c_2\| \leqslant d\}$. Equivalently we have $D = (c_1 + S') \cap (c_2 + S')$. If S was a unit disk, D would be the intersection of two disks with radius d, such that the boundary of one contains the center of the other.

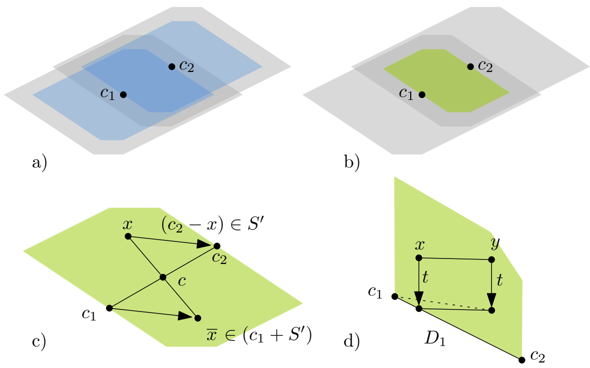

Figure 5 a) The gray sets are scaled about their center so that the center of one set is on the boundary of the other. b) the intersection D. c) Illustration of Lemma 15. d) Illustration of Lemma 17.

▶ **Lemma 15.** *The set D is centrally symmetric around $c := (c_1 + c_2)/2$.*

Proof. Let x be a point in D, we need to show that $\bar{x} := x + 2(c - x)$ is in D too. As $D = (c_1 + S') \cap (c_2 + S')$, it is sufficient to show $\bar{x} \in c_1 + S'$ and $\bar{x} \in c_2 + S'$. By definition, $\bar{x}$ is equal to $c_1 + c_2 - x$. Since x is in D, then $\|c_2 - x\| \leqslant d$, which implies that $c_2 - x$ is in S'. Therefore $\bar{x}$ is in $c_1 + S'$. By the symmetry of the arguments, we obtain that $\bar{x}$ is in D. ◀

▶ **Lemma 16.** *The tangents to D at c_1 and c_2 are parallel.*

Proof. Let us denote by ℓ_1 the tangent to D at c_1. Then we denote by ℓ_2 the line parallel to ℓ_1 that contains c_2. We claim that ℓ_2 is tangent to D. By construction D is convex, as the intersection of two convex sets. This implies that ℓ_2 is tangent to D if and only if $D \cap \ell_2$ is a line segment that contains c_2. This line segment may be only one point. Let x be a point in $D \cap \ell_2$. By Lemma 15, D is centrally symmetric around c. Therefore $x + 2(c - x)$ is in D, and by construction it is also in ℓ_1. Since $D \cap \ell_1$ is a line segment that contains c_1, thus $D \cap \ell_2$ is a line segment that contains c_2. ◀

We cut D along the line ℓ going through c_1 and c_2, and split D into two sets denoted by D_1 and D_2. We define D_1 as the set of points below this line, and D_2 as the set of points not below.

▶ **Lemma 17.** *Let i be in $\{1, 2\}$, and let x and y be in D_i. Then we have $\|x - y\| \leqslant d$.*

Proof. We do the proof for $i = 1$, and the case $i = 2$ can be done symmetrically. By Lemma 16, the tangents ℓ_1 and ℓ_2 of D at c_1 and c_2 are parallel. Without loss of generality, let us assume that they are vertical, that c_1 is to the left of c_2 and x to the left of y. We denote by $\tilde{x}$ (respectively $\tilde{y}$) the vertical projection of x (respectively y) on ℓ. Without loss of generality $\|x - \tilde{x}\| \leqslant \|y - \tilde{y}\|$. We define $t = x - \tilde{x}$. Note that $\|x - y\| = \|(x - t) - (y - t)\| = \|\tilde{x} - (y - t)\|$. Furthermore, we can move $\tilde{x}$ on ℓ towards c_1 and this will only increase the distance to $(y - t)$. We get $\|\tilde{x} - (y - t)\| \leqslant \|c_1 - (y - t)\|$. By definition $(y - t) \in D_1 \subset D$ and thus $\|c_1 - (y - t)\| \leqslant d$. This implies $\|x - y\| \leqslant d$ and finishes the proof. ◀

Following the arguments of Clark et al. [19], one first guesses in quadratic time S_1 and S_2 in a maximum clique C such that the distance between their centers $\|c_1 - c_2\|$ is maximized among the pairs $S_1, S_2 \in C$. One can then remove all the objects not centered in D. By Lemma 17, the intersection graph induced by the sets centered in D is cobipartite. Since computing an independent set in a bipartite graph can be done in polynomial time, then one can compute a maximum clique in G in polynomial time.

Before explaining how to compute a maximum clique when no representation is given, we need to introduce a few definitions. Let $\Lambda = e_1, e_2, \ldots, e_m$ be an ordering of the m edges of G. Let $G_\Lambda(k)$ be the subgraph of G with edge-set $\{e_k, e_{k+1}, \ldots, e_m\}$. For each $e_k = uv$, $N_{\Lambda,k}$ is defined as the set of vertices adjacent to u and v in $G_\Lambda(k)$.

▶ **Definition 18** (Raghavan and Spinrad [46]). An edge ordering $\Lambda = e_1, e_2, \ldots, e_m$ is a *cobipartite neighborhood edge elimination ordering* (CNEEO), if for each e_k, $N_{\Lambda,k}$ induces a cobipartite graph in G.

Proof of Theorem 13. Raghavan and Spinrad have given a polynomial time algorithm that takes an abstract graph as input, and returns a CNEEO or a certificate that no CNEEO exists for the graph. Secondly, they showed how to compute in polynomial time a maximum clique when given a graph and a CNEEO on it. Therefore, it is sufficient to show that for any centrally symmetric, bounded, convex set S, and any intersection graph G of translated of S, there exists a CNEEO on G. Let us consider such a graph G with a representation. Arguing with Lemma 17 as previously, ordering the edges by non-increasing length gives a CNEEO, where the length of an edge is the distance between the two centers. ◀

References

1 Anders Aamand, Mikkel Abrahamsen, Jakob B. T. Knudsen, and Peter M. R. Rasmussen. Classifying convex bodies by their contact and intersection graphs. *CoRR*, abs/1902.01732, 2019. arXiv:1902.01732.

2 Akanksha Agrawal, Grzegorz Guspiel, Jayakrishnan Madathil, Saket Saurabh, and Meirav Zehavi. Connecting the dots (with minimum crossings). In *35th International Symposium on Computational Geometry, SoCG 2019, June 18-21, 2019, Portland, Oregon, USA.*, pages 7:1–7:17, 2019. doi:10.4230/LIPIcs.SoCG.2019.7.

3 Paola Alimonti and Viggo Kann. Some APX-completeness results for cubic graphs. *Theor. Comput. Sci.*, 237(1-2):123–134, 2000. doi:10.1016/S0304-3975(98)00158-3.

4 Boris Aronov, Anirudh Donakonda, Esther Ezra, and Rom Pinchasi. On pseudo-disk hypergraphs. *arXiv preprint arXiv:1802.08799*, 2018.

5 Piotr Berman and Marek Karpinski. Efficient amplifiers and bounded degree optimization. *Electronic Colloquium on Computational Complexity (ECCC)*, 8(53), 2001. URL: http://eccc.hpi-web.de/eccc-reports/2001/TR01-053/index.html.

6 Marthe Bonamy, Édouard Bonnet, Nicolas Bousquet, Pierre Charbit, and Stéphan Thomassé. EPTAS for max clique on disks and unit balls. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 568–579, 2018. doi:10.1109/FOCS.2018.00060.

7 Édouard Bonnet, Panos Giannopoulos, Eun Jung Kim, Paweł Rzążewski, and Florian Sikora. QPTAS and subexponential algorithm for maximum clique on disk graphs. In *34th International Symposium on Computational Geometry, SoCG 2018, June 11-14, 2018, Budapest, Hungary*, pages 12:1–12:15, 2018. doi:10.4230/LIPIcs.SoCG.2018.12.

8 Édouard Bonnet, Nicolas Grelier, and Tillmann Miltzow. Maximum clique in disk-like intersection graphs. *arXiv preprint arXiv:2003.02583*, 2020.

9 Édouard Bonnet and Tillmann Miltzow. Parameterized hardness of art gallery problems. In *24th Annual European Symposium on Algorithms, ESA 2016, August 22-24, 2016, Aarhus, Denmark*, pages 19:1–19:17, 2016. doi:10.4230/LIPIcs.ESA.2016.19.

10 Édouard Bonnet and Paweł Rzążewski. Optimality program in segment and string graphs. *Algorithmica*, 81(7):3047–3073, 2019. doi:10.1007/s00453-019-00568-7.

11 Andreas Brandstädt, Van Bang Le, and Jeremy P. Spinrad. *Graph classes: a survey.* SIAM, 1999.

12 Heinz Breu and David G. Kirkpatrick. Unit disk graph recognition is NP-hard. *Comput. Geom.*, 9(1-2):3–24, 1998. doi:10.1016/S0925-7721(97)00014-X.

13 Valentin E. Brimkov, Konstanty Junosza-Szaniawski, Sean Kafer, Jan Kratochvíl, Martin Pergel, Paweł Rzążewski, Matthew Szczepankiewicz, and Joshua Terhaar. Homothetic polygons and beyond: Maximal cliques in intersection graphs. *Discrete Applied Mathematics*, 247:263–277, 2018. doi:10.1016/j.dam.2018.03.046.

14 Ayelet Butman, Danny Hermelin, Moshe Lewenstein, and Dror Rawitz. Optimization problems in multiple-interval graphs. *ACM Trans. Algorithms*, 6(2):40:1–40:18, 2010. doi:10.1145/1721837.1721856.

15 Sergio Cabello, Jean Cardinal, and Stefan Langerman. The clique problem in ray intersection graphs. *Discrete & Computational Geometry*, 50(3):771–783, 2013. doi:10.1007/s00454-013-9538-5.

16 Paz Carmi, Matthew J. Katz, and Pat Morin. Stabbing pairwise intersecting disks by four points. *CoRR*, abs/1812.06907, 2018. arXiv:1812.06907.

17 Jérémie Chalopin and Daniel Gonçalves. Every planar graph is the intersection graph of segments in the plane: extended abstract. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 631–638, 2009. doi:10.1145/1536414.1536500.

18 Timothy M. Chan. Polynomial-time approximation schemes for packing and piercing fat objects. *J. Algorithms*, 46(2):178–189, 2003. doi:10.1016/S0196-6774(02)00294-8.

19 Brent N. Clark, Charles J. Colbourn, and David S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86(1-3):165–177, 1990. `doi:10.1016/0012-365X(90)90358-O`.

20 Ludwig Danzer. Zur lösung des gallaischen problems über kreisscheiben in der euklidischen ebene. *Studia Sci. Math. Hungar*, 21(1-2):111–134, 1986.

21 Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012.

22 Y. G. Dorfman and G. I. Orlova. Finding the maximal cut in a graph. *Engineering Cybernetics*, 10:502–506, 1972.

23 Thomas Erlebach, Klaus Jansen, and Eike Seidel. Polynomial-time approximation schemes for geometric intersection graphs. *SIAM J. Comput.*, 34(6):1302–1323, 2005. `doi:10.1137/S0097539702402676`.

24 Aleksei V. Fishkin. Disk graphs: A short survey. In *Approximation and Online Algorithms, First International Workshop, WAOA 2003, Budapest, Hungary, September 16-18, 2003, Revised Papers*, pages 260–264, 2003. `doi:10.1007/978-3-540-24592-6_23`.

25 Mathew C. Francis, Daniel Gonçalves, and Pascal Ochem. The Maximum Clique Problem in Multiple Interval Graphs. *Algorithmica*, 71(4):812–836, 2015. `doi:10.1007/s00453-013-9828-6`.

26 Grzegorz Guspiel. Complexity of finding perfect bipartite matchings minimizing the number of intersecting edges. *CoRR*, abs/1709.06805, 2017. `arXiv:1709.06805`.

27 Sariel Har-Peled, Haim Kaplan, Wolfgang Mulzer, Liam Roditty, Paul Seiferth, Micha Sharir, and Max Willert. Stabbing pairwise intersecting disks by five points. In *29th International Symposium on Algorithms and Computation, ISAAC 2018, December 16-19, 2018, Jiaoxi, Yilan, Taiwan*, pages 50:1–50:12, 2018. `doi:10.4230/LIPIcs.ISAAC.2018.50`.

28 Johan Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, 2001. `doi:10.1145/502090.502098`.

29 Petr Hlinený and Jan Kratochvíl. Representing graphs by disks and balls (a survey of recognition-complexity results). *Discrete Mathematics*, 229(1-3):101–124, 2001. `doi:10.1016/S0012-365X(00)00204-1`.

30 Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-SAT. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001. `doi:10.1006/jcss.2000.1727`.

31 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, December 2001.

32 Ross J. Kang and Tobias Müller. Sphere and Dot Product Representations of Graphs. *Discrete & Computational Geometry*, 47(3):548–568, 2012. `doi:10.1007/s00454-012-9394-8`.

33 Seog-Jin Kim, Alexandr Kostochka, and Kittikorn Nakprasit. On the chromatic number of intersection graphs of convex sets in the plane. *the electronic journal of combinatorics*, 11(1):52, 2004.

34 Paul Koebe. Kontaktprobleme der konformen Abbildung. *Berichte über die Verhandlungen der Sächsischen Akademie der Wissenschaften zu Leipzig, Mathematisch-Physikalische Klasse*, 88:141–164, 1936.

35 Jan Kratochvíl. String graphs. II. recognizing string graphs is NP-hard. *J. Comb. Theory, Ser. B*, 52(1):67–78, 1991. `doi:10.1016/0095-8956(91)90091-W`.

36 Jan Kratochvíl and Jiří Matoušek. Intersection graphs of segments. *J. Comb. Theory, Ser. B*, 62(2):289–315, 1994. `doi:10.1006/jctb.1994.1071`.

37 Dániel Marx. Parameterized complexity of independence and domination on geometric graphs. In *Parameterized and Exact Computation, Second International Workshop, IWPEC 2006, Zürich, Switzerland, September 13-15, 2006, Proceedings*, pages 154–165, 2006. `doi:10.1007/11847250_14`.

38 Dániel Marx and Michal Pilipczuk. Optimal parameterized algorithms for planar facility location problems using voronoi diagrams. In *Algorithms - ESA 2015 - 23rd Annual European*

Symposium, Patras, Greece, September 14-16, 2015, Proceedings, pages 865–877, 2015. doi:10.1007/978-3-662-48350-3_72.

39 Terry A. McKee and Fred R. McMorris. *Topics in intersection graph theory*. SIAM, 1999.

40 Matthias Middendorf and Frank Pfeiffer. The max clique problem in classes of string-graphs. *Discrete Mathematics*, 108(1-3):365–372, 1992. doi:10.1016/0012-365X(92)90688-C.

41 Bernard M. E. Moret. Planar NAE3SAT is in P. *ACM SIGACT News*, 19(2):51–54, 1988.

42 Dana Moshkovitz and Ran Raz. Sub-constant error probabilistically checkable proof of almost-linear size. *Computational Complexity*, 19(3):367–422, 2010. doi:10.1007/s00037-009-0278-0.

43 Tim Nieberg and Johann Hurink. A PTAS for the minimum dominating set problem in unit disk graphs. In *Approximation and Online Algorithms, Third International Workshop, WAOA 2005, Palma de Mallorca, Spain, October 6-7, 2005, Revised Papers*, pages 296–306, 2005. doi:10.1007/11671411_23.

44 Tim Nieberg, Johann Hurink, and Walter Kern. A robust PTAS for maximum weight independent sets in unit disk graphs. In *Graph-Theoretic Concepts in Computer Science, 30th International Workshop, WG 2004, Bad Honnef, Germany, June 21-23, 2004, Revised Papers*, pages 214–221, 2004. doi:10.1007/978-3-540-30559-0_18.

45 Svatopluk Poljak. A note on stable sets and colorings of graphs. *Commentationes Mathematicae Universitatis Carolinae*, 15(2):307–309, 1974.

46 Vijay Raghavan and Jeremy P. Spinrad. Robust algorithms for restricted domains. *J. Algorithms*, 48(1):160–172, 2003. doi:10.1016/S0196-6774(03)00048-8.

47 Marcus Schaefer, Eric Sedgwick, and Daniel Stefankovic. Recognizing string graphs in NP. *J. Comput. Syst. Sci.*, 67(2):365–380, 2003. doi:10.1016/S0022-0000(03)00045-X.

48 Lajos Stachó. A solution of gallai's problem on pinning down circles. *Mat. Lapok*, 32(1-3):19–47, 1981.

49 Erik Jan van Leeuwen. Better approximation schemes for disk graphs. In *Algorithm Theory - SWAT 2006, 10th ScandinavianWorkshop on Algorithm Theory, Riga, Latvia, July 6-8, 2006, Proceedings*, pages 316–327, 2006. doi:10.1007/11785293_30.

50 Erik Jan van Leeuwen. *Optimization and Approximation on Systems of Geometric Objects*. PhD thesis, Utrecht University, 2009.