Independence and coloring problems on intersection graphs of disks

Thomas Erlebach[1] and Jiří Fiala[⋆2]

[1] ETH Zürich, Computer Engineering and Networks Lab,
CH-8092 Zürich, Switzerland.
E-mail: erlebach@tik.ee.ethz.ch

[2] Charles University, Faculty of Mathematics and Physics,
Institute for Theoretical Computer Science (ITI)[⋆⋆⋆],
Malostranské nám. 2/25, 118 00, Prague, Czech Republic.
E-mail: fiala@kam.mff.cuni.cz

Abstract. This paper surveys on-line and approximation algorithms for the maximum independent set and coloring problems on intersection graphs of disks. As a new result, it is shown that no deterministic on-line algorithm can achieve competitive ratio better than $\Omega(\log n)$ for disk graphs and for square graphs with n vertices, even if the geometric representation is given as part of the input. Furthermore, it is proved that the standard First-fit heuristic achieves competitive ratio $O(\log n)$ for disk graphs and for square graphs and is thus best possible.

1 Introduction

The class of intersection graphs of disks in the Euclidean plane, called disk graphs, was studied for many years for its theoretical aspects as well as for its applications. As an example of a classical result we mention the theorem of Koebe who proved in 1936 that every planar graph can be represented as a coin graph, i.e. a disk graph where disks are not allowed to overlap [19] (see also the more accessible discussion of Koebe's result by Sachs [26]).

In contrast to the case of planar graphs, no efficient methods are known for the recognition of disk graphs. Breu and Kirkpatrick have shown that the recognition problem is *NP*-hard for unit disk graphs (intersection graphs of disks with equal diameter) [4] and for disk graphs with bounded diameter ratio (intersection graphs of disks where the ratio

⋆ This author was partially supported by EU APPOL project IST-1999-14084, during his postdoctoral fellowship at Christian-Albrechts University of Kiel and by NATO science fellowship, administered through Norwegian Research Council project no. 143192/410 during his postdoctoral fellowship at University of Bergen.

⋆⋆⋆ Supported by the Ministry of Education of the Czech Republic as project LN00A056.

of the largest diameter to the smallest diameter is bounded by an arbitrary constant) [3]. Hliněný and Kratochvíl proved NP-hardness for the recognition problem of arbitrary disk graphs [13].

The hardness of the recognition problem implies that a disk representation cannot be derived from the graph in polynomial time unless $P = NP$. Therefore, an important factor in the design of algorithms for disk graphs is whether the disk graph is given only as a set of edges and vertices, or whether the centers and radii of the disks (called the *disk representation* of the graph) form the input to the algorithm. Some problems can be solved efficiently no matter whether the disk representation is given or not. The problem of computing a maximum clique in a unit disk graph is an example: Raghavan and Spinrad presented an efficient algorithm that does not require the disk representation [25].

The maximum independent set problem on disk graphs (computing a largest subset of the given disks such that the disks in the subset are pairwise disjoint) has applications in map labeling. Under the assumption that labels occupy a circular area, the maximum number of non-intersecting labels that can be placed on a map (out of a given set of desired labels) is equal to the size of the maximum independent set in the corresponding disk graph. For applications in map labeling, it is clearly interesting to extend the problem to rectangular labels (representing text), sliding labels, etc. For a bibliography on map labeling problems we refer to the on-line web catalogue maintained by Wolff and Strijk [28].

One of the most practical applications of disk graphs was mentioned by Hale, who pointed out in 1980 that the channel assignment problem can be modeled as a graph theoretical problem, if we assume that all transmitters have circular range and transmitters with intersecting ranges are to use different frequencies [12]. Clearly, the underlying graph is a disk graph, and the problem is equivalent to the graph coloring problem. Observe also that in this case we may assume that the disk representation can be derived from the placement of transmitters and their ranges.

We focus our study on approximation and on-line algorithms for the maximum independent set problem and the coloring problem for intersection graphs of disks. In the first part (Sections 2 and 3) we survey results for both problems on disk graphs and two subclasses: unit disk graphs and disk graphs with bounded diameter ratio. We provide known upper and lower bounds on the approximation and competitive ratios, and we also discuss the impact if the disk representation is given as part of the input. Then, in Section 4, we present new results concerning on-line coloring of disk graphs and disk graphs with bounded diameter ratio. As a supplemental result we solve the corresponding problem on squares as well.

1.1 Preliminaries

A *disk* in the Euclidean plane is specified by its center and its diameter. We denote the center of a disk D by c_D. Given a set of disks, the *intersection graph* of the disks is the graph with one vertex for each disk and with an edge between two vertices if the corresponding disks have a non-empty intersection. Throughout this paper we consider only closed disks. Therefore, tangent disks are considered as intersecting. A graph G is called a *disk graph* if there exists a set of disks such that G is their intersection graph. Such a set of disks is called a *disk representation*, *disk model*, or *geometric representation* of G. A graph is called a *unit disk graph* if it is the intersection graph of a set of disks with the same diameter (w.l.o.g., we assume that the diameter is 1 in this case). The *diameter ratio* of a set of disks is the ratio of the maximum diameter of a disk to the smallest diameter. Intersection graphs of disks whose diameter ratio is bounded by σ are called *σ-bounded disk graphs.*

These notions extend to the intersection graphs of other geometric objects in a straightforward way. In particular, we will also consider *square graphs*, i.e., the intersection graphs of squares whose sides are parallel to the coordinate axes.

For a given graph $G = (V, E)$, the set of neighbors of a vertex $v \in V$ is denoted by $N(v) = \{u \in V : \{v, u\} \in E\}$. A subset $I \subseteq V$ is an *independent set* if the vertices in I are mutually non-adjacent. A subset $C \subseteq V$ is a *clique* if the vertices in C are pairwise adjacent. The maximum independent set problem and the maximum clique problem are the problems of computing a largest independent set and a largest clique, respectively. A *coloring* of G is an assignment of colors to vertices such that adjacent vertices receive different colors. The (minimum) coloring problem is the problem of computing a coloring using as few distinct colors as possible. These problems are notoriously hard to approximate on general graphs, but are often easier to approximate (or even to solve optimally) on restricted classes of graphs.

For an optimization problem, an *approximation algorithm* computes a feasible solution in time polynomial in the size of the input. It has *approximation ratio* ρ if for every input, the value of the computed solution is at most ρ times the optimum (for a minimization problem) or at least $1/\rho$ times the optimum (for a maximization problem). A *polynomial-time approximation scheme* is an algorithm that, given an instance of the problem and a parameter $\varepsilon > 0$, computes a feasible solution that is at most a factor of $1 + \varepsilon$ away from the optimum and whose running-time is polynomial in the size of the instance for every fixed $\varepsilon > 0$.

An on-line algorithm receives the vertices of the graph (or the disks) one by one together with the incident edges connecting the current vertex to previously presented vertices. It must decide the solution for the current vertex (membership in the independent set or the color assigned to the vertex) immediately without knowledge of future vertices and edges. An on-line algorithm achieves *competitive ratio* ρ if it always produces a solution that is at most a factor of ρ away from the optimum. Such an algorithm is called ρ-*competitive*.

2 The maximum independent set problem

The maximum independent set problem has been proved *NP*-complete for unit disk graphs even if the disk representation is given [27, 6]. Of course, this implies that the problem is *NP*-complete for σ-bounded disk graphs and general disk graphs as well, and also in the case when the representation is not given. Therefore, one is interested in approximation algorithms for the problem.

2.1 Independent sets in unit disk graphs

A first natural algorithm to consider is the greedy algorithm. It starts with an empty set $I = \emptyset$ and then processes the disks in arbitrary order. If the current disk is disjoint from all disks in I, it is added to I. When all disks have been processed, the set I is output. It is easy to see that the greedy algorithm does not require the disk representation and that it is an on-line algorithm.

Theorem 1. *The greedy algorithm is a* 5*-competitive on-line algorithm for the maximum independent set problem in unit disk graphs. It does not require the disk representation.*

Proof. Consider some optimal solution I^*. Whenever the greedy algorithm accepts a disk D_i, remove from I^* all disks that intersect D_i (including D_i itself). It is clear that I^* is empty at the end of the greedy algorithm. Furthermore, we claim that each disk accepted by the greedy algorithm removes at most 5 disks from I^*, thus establishing that the competitive ratio is at most 5. Assume that the greedy algorithm accepts a disk D_i. If D_i is also contained in I^*, it suffices to remove one disk from

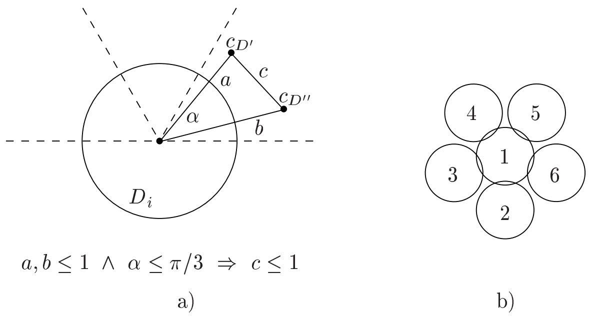

Fig. 1. Exploring the neighborhood of a unit disk.

I^*. If D_i is not contained in I^*, then all disks in I^* that intersect D_i have to be removed. There can be five such disks, as shown in Fig. 1 b). There cannot be six such disks, however: for any two disks D' and D'' that intersect D_i and that do not intersect each other, the angle between $\overline{c_{D_i} c_{D'}}$ and $\overline{c_{D_i} c_{D''}}$ must be larger than $\pi/3$, as illustrated in Fig. 1 a), and if there were six pairwise non-overlapping disks intersecting D_i, the sum of the angles between them would be larger than 2π, a contradiction. $\square$

Furthermore, no on-line algorithm can have competitive ratio better than 5 on unit disk graphs even if the disk representation is given: Assume that the input consists of the six disks in Fig. 1 b) and that the disk in the middle is presented first. Any on-line algorithm must accept the first disk (otherwise, it will have unbounded ratio on an instance consisting of this disk only) and will hence reject the other five disks, while the optimal independent set consists of these five disks.

As shown by Marathe et al. [22], the approximation ratio of the greedy algorithm can be improved from 5 to 3 in the off-line case by processing the given disks in order of non-decreasing y-coordinates of their centers. The argument in the proof of Theorem 1 can be adapted to show that among the disks that intersect the current disk D_i and that are processed later, there can be at most three pairwise disjoint disks, see again Fig. 1 a). Thus, at most three disks have to be removed from I^* for each disk accepted by the greedy algorithm.

This 3-approximation algorithm can be adapted to the case where the disk representation is not given as follows. A vertex whose neighborhood $N(v)$ does not contain an independent set of size larger than three can be found in polynomial time (e.g., in time $O(|V|^5)$ by enumerating all 4-element subsets of $N(v)$ for each $v \in V$). Such a vertex exists because the disk with lowest y-coordinate satisfies the property. Once such a vertex is determined, it is added to the independent set, and the vertex and all its neighbors are removed from the graph. This is repeated until the graph is empty.

Theorem 2 (Marathe et al. [22]). *There is a 3-approximation algorithm for the maximum independent set problem in unit disk graphs that does not require the disk representation.*

For the case that the disk representation is given, a polynomial-time approximation scheme can be obtained using the shifting technique invented by Baker [2] and Hochbaum and Maass [14]. Such an approximation scheme was presented by Hunt III et al. [15] and by Matsui [23].

We sketch the basic ideas. Let a set $\mathcal{D}$ of disks be given. Denote by I^* some optimal independent set. Without loss of generality assume that the given disks have diameter 1 and that no center has an integral coordinate. Consider a grid consisting of horizontal and vertical lines at all integer coordinates. Fix some integer $k > 0$. For each pair of integers (i, j) such that $0 \leq i, j \leq k-1$, consider the subset $\mathcal{D}_{i,j}$ of disks obtained by removing all disks that intersect a vertical line at $x = i + kp$ for some $p \in \mathbb{Z}$ or some horizontal line at $y = j + kp$ for some $p \in \mathbb{Z}$. See Fig. 2 for an example. If the vertical lines at $x = i + kp$, $p \in \mathbb{Z}$, and the horizontal lines at $y = j + kp$, $p \in \mathbb{Z}$, are removed from the plane, disjoint open squares with side length k and area k^2 remain. Each disk in $\mathcal{D}_{i,j}$ is completely contained in one such square. Therefore, a maximum independent set of $\mathcal{D}_{i,j}$ is the union of maximum independents sets of disks in all squares. A maximum independent set among the disks in one square can contain at most $O(k^2)$ disks, and can hence be computed in time $|\mathcal{D}|^{O(k^2)}$ by enumeration of all subsets of size $O(k^2)$. Thus, a maximum independent set of $\mathcal{D}_{i,j}$ can be computed in polynomial time for fixed k. The algorithm computes a maximum independent set of $\mathcal{D}_{i,j}$ for all k^2 pairs (i, j) and outputs the largest among these sets as the solution. The approach is called *shifting* because trying all possible values of i and j can be viewed as shifting the grid through the plane.

The cardinality of the solution output by the algorithm is at least $(1 - \frac{2}{k})|I^*|$. To see this, note that each disk intersects only one horizontal line at an integer coordinate and one vertical line, respectively. Hence, there exists a value of i such that at most $|I^*|/k$ disks in I^* intersect vertical lines $x = i + kp$ ($p \in \mathbb{Z}$). Similarly, there is a value of j such that at most $|I^*|/k$ disks in I^* intersect horizontal lines $y = j + kp$ ($p \in \mathbb{Z}$). Thus, the set $\mathcal{D}_{i,j}$ for these values of i and j still contains an independent set of size at least $(1 - \frac{2}{k})|I^*|$. Since the algorithm computes a maximum

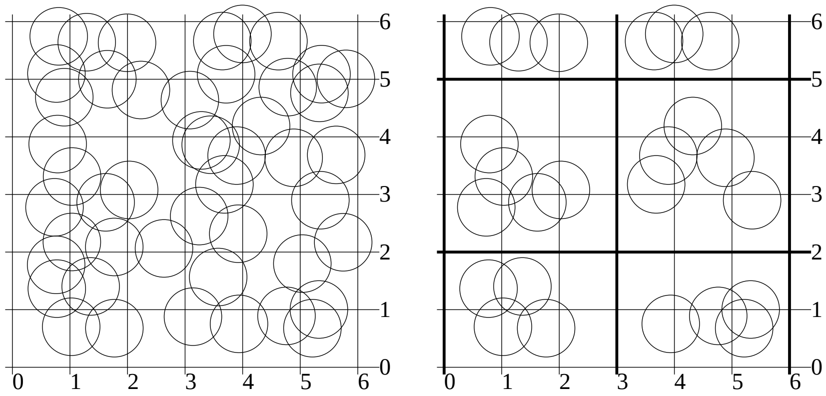

Fig. 2. Illustration of the shifting strategy for $k = 3$ and the choice $i = 0$ and $j = 2$. The given disks are shown on the left-hand side. If all disks intersecting a vertical line at $x = 3p$ for $p \in \mathbb{Z}$ or a horizontal line at $y = 2 + 3p$ for $p \in \mathbb{Z}$ (drawn in bold) are removed, the disks shown on the right-hand side remain and constitute $\mathcal{D}_{0,2}$.

independent set in each $\mathcal{D}_{i,j}$, the largest such set must have cardinality at least $(1 - \frac{2}{k})|I^*|$.

For a given $\varepsilon > 0$, we can thus choose $k = \lceil 2/\varepsilon \rceil$ to obtain a solution of size at least $(1-\varepsilon)|I^*|$. The running-time is $|\mathcal{D}|^{O(k^2)}$. It is shown in [15, 23] that the running-time can be reduced to $|\mathcal{D}|^{O(k)}$ by removing only those disks that intersect a horizontal grid line at $y = j + kp$ for some $p \in \mathbb{Z}$ and then using dynamic programming to compute an optimal independent set in each strip of width k between two of these horizontal grid lines.

Theorem 3 (Hunt III et al. [15], Matsui [23]). *There is a polynomial-time approximation scheme for the maximum independent set problem in unit disk graphs provided that the disk representation is given as part of the input.*

It remains an open problem whether the maximum independent set problem in unit disk graphs admits a polynomial-time approximation scheme also in the case where the representation is not given as part of the input.

2.2 Independent sets in general disk graphs

In general disk graphs, the greedy algorithm can have approximation ratio $n - 1$ for instances with n disks: An instance could consist of one disk D (presented first) and $n - 1$ smaller disks intersecting D, but not intersecting each other. The greedy algorithm would accept only the first disk, but the optimal solution would consist of the $n-1$ other disks. This instance also shows that no deterministic on-line algorithm can achieve competitive ratio better than $n-1$, even if the disk representation is given.

Theorem 4. *The greedy algorithm is a $(n-1)$-competitive algorithm for the maximum independent set problem in disk graphs. It works even if the disk representation is not given. No deterministic on-line algorithm can have competitive ratio smaller than $n-1$, even if the disk representation is given.*

In the off-line case, the idea of the 3-approximation algorithm for unit disk graphs of Theorem 2 can be adapted to give a 5-approximation algorithm for general disk graphs, as shown by Marathe et al. [22]. If the disk representation is given, it suffices to sort the disks in order of non-decreasing diameters and apply the greedy algorithm to this order. For each disk D_i, the set of disks that are processed later and that intersect D_i contains at most five pairwise disjoint disks. This follows by the same argument as in the proof of Theorem 2, because all disks that are processed later are at least as big as D_i. Thus, the greedy algorithm achieves approximation ratio 5 for disk graphs if it processes the disks in order of non-decreasing diameters. Again, the algorithm can be adapted to the case where the disk representation is not available: It suffices to identify a vertex whose neighborhood $N(v)$ does not contain an independent set of size at least 6. Such a vertex must exist, and it can be found in polynomial time.

Theorem 5 (Marathe et al. [22]). *There is a 5-approximation algorithm for the maximum independent set problem in disk graphs that does not require the disk representation.*

For the case that the disk representation is part of the input, a polynomial-time approximation scheme for the maximum independent set problem in disk graphs has been devised by Erlebach, Jansen and Seidel [7]. It is also based on the shifting technique, but the given disks are first partitioned into layers according to their diameters, and the shifting strategy is applied on all layers simultaneously, using grids of different granularity on different layers. For each choice of the shifting parameters i and j, $0 \leq i, j \leq k-1$, the optimal independent set in $\mathcal{D}_{i,j}$ is then computed using dynamic programming, beginning at the layer that contains the smallest disks.

Theorem 6 (Erlebach, Jansen and Seidel [7]). *There is a polynomial-time approximation scheme for the maximum independent set problem in disk graphs provided that the disk representation is given as part of the input.*

2.3 Independent sets in bounded disk graphs

Now consider σ-bounded disk graphs. Assume that the minimum diameter of a disk is equal to 1 and, consequently, the maximum diameter is at most σ. In the off-line case, we can just apply the algorithms for general disk graphs and get a 5-approximation algorithm for the case without given representation and a polynomial-time approximation scheme for the case with given representation.

In the on-line case, the competitive ratio of the greedy algorithm can be bounded by $O(\min\{\sigma^2, n\})$ for σ-bounded disk graphs. To see this, again let I^* denote some optimal independent set. When the algorithm accepts a disk D, we remove all neighbors of D (i.e., all disks intersecting D) from I^*. We claim that the neighborhood of D can contain at most $O(\sigma^2)$ disjoint disks. To see this, note that all neighbors of D must have their center within distance σ from the center of D. Thus, all neighbors of D are contained in a circle with radius 1.5σ around the center of D. This circle has area $2.25\sigma^2\pi$. Each disk has diameter at least 1 and thus occupies an area of at least $\pi/4$. This shows that the neighborhood of D can contain at most $9\sigma^2$ disjoint disks. So, for each disk accepted by the greedy algorithm, we have to remove at most $9\sigma^2$ disks from I^*. Thus, we get an upper bound of $O(\sigma^2)$ for the competitive ratio of the greedy algorithm. An upper bound of $n-1$ is trivial.

To show that no deterministic on-line algorithm can do better than the greedy algorithm in the worst case, consider an instance consisting of one disk with diameter σ followed by $\Theta(\sigma^2)$ disks of diameter 1 that intersect σ and that are disjoint from each other. On this instance, the solution of the greedy algorithm is a factor of $n-1 = \Theta(\sigma^2)$ smaller than the optimal solution.

Theorem 7. *The greedy algorithm achieves competitive ratio $O(\min\{n, \sigma^2\})$ for σ-bounded disk graphs with n disks. It does not require the disk representation. Every deterministic algorithm has competitive ratio $\Omega(\min\{n, \sigma^2\})$ even if the disk representation is given.*

3 The coloring problem

The problem of deciding whether a unit disk graph with given representation can be colored with three colors has been shown *NP*-complete by Clark, Colbourn and Johnson [6] using a reduction from 3-colorability of planar graphs with maximum degree 3. This implies that there cannot be an approximation algorithm for the coloring problem on unit disk graphs with approximation ratio smaller than 4/3.

As in the case of the maximum independent set problem, the *NP*-completeness for unit disk graphs with given representation implies *NP*-completeness of deciding 3-colorability of disk graphs or σ-bounded disk graphs, and also for the variants without given representation.

It was proved by Gräf et al. in [10] that deciding k-colorability remains *NP*-complete for unit disk graphs for any fixed number $k \geq 3$.

3.1 Coloring unit disk graphs

First-fit is one of the most well-known heuristics for on-line graph coloring. It processes the vertices of the given graph in some order and assigns each vertex the smallest available color, i.e., the smallest color that has not yet been assigned to an adjacent vertex. We assume that colors are represented by positive integers. If at most k vertices adjacent to v have been colored prior to v, the color assigned to v by First-fit is contained in $\{1, 2, \ldots, k+1\}$, because at most k of these colors can already have been assigned to neighbors of v.

Let us apply the First-fit coloring algorithm to unit disk graphs. Consider some disk D at the time it is assigned its color. Let $d(D)$ denote the number of intersecting disks that have been colored before. The color assigned to D is at most $d(D) + 1$. On the other hand, the closed neighborhood of D (the set of all disks intersecting D, including D itself) does not contain an independent set of size larger than 5. Thus, at most 5 of the disks in the closed neighborhood of D can be assigned the same color in any coloring. Therefore, even the optimal coloring must use at least $(d(D) + 1)/5$ colors. This shows that First-fit is a 5-approximation algorithm for coloring unit disk graphs. Furthermore, First-fit is an on-line algorithm that does not need the disk representation.

Theorem 8. *First-fit is an on-line algorithm with competitive ratio at most* 5 *for unit disk graphs. It does not require the disk representation.*

A lower bound of 2 on the competitive ratio of any on-line coloring algorithm for unit disk graphs was presented by Fiala et al. in [8]. We reproduce here the counterexample showing that no on-line algorithm can be $(2-\varepsilon)$-competitive for any $\varepsilon > 0$. Consider the graph and its representation depicted in Fig. 3 and order the disks as indicated by the numbers. Assume that there is an algorithm with competitive ratio $2-\varepsilon$. The vertices 1–6 form an independent set and must be colored with the same color by the algorithm, because otherwise its competitive ratio would be at least 2 on these 6 vertices. On vertices 7–12 the algorithm may use two new colors (but no more, if its competitive ratio is to be smaller than 2). Then, however, it will need three extra colors for the central triple, thus using 6 colors in total. Since the optimal coloring uses only 3 colors, this contradicts the assumption that the algorithm is $(2-\varepsilon)$-competitive.

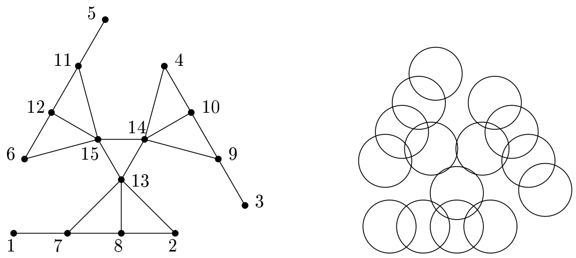

Fig. 3. A difficult instance for on-line unit disk coloring.

Theorem 9 (Fiala et al. [8]). *No deterministic algorithm can achieve competitive ratio smaller than* 2 *for on-line coloring of unit disk graphs, even if the disk representation is part of the input.*

In the off-line case, 3-approximation algorithms for coloring unit disk graphs were presented by Peeters [24] and Marathe et al. [22]. If the disk representation is given, apply the First-fit algorithm to the disks in the order of non-increasing y-coordinates of their centers. Consider a disk D at the time it is assigned its color. The $d(D)$ previously colored disks that intersect D have a y-coordinate that is not smaller than the y-coordinate of D. As in the discussion leading to Theorem 2 (see also Fig. 1), the set of disks containing D and its previously colored neighbors does not have an independent set of size larger than 3. Therefore, even in an optimal coloring $(d(D)+1)/3$ colors are required just for these disks. Since the color assigned to D is at most $d(D)+1$, approximation ratio 3 is achieved.

To adapt the algorithm to the case without given representation, we can simply order the vertices in the recursively defined smallest-degree-last order and apply the First-fit algorithm. This means that we select some vertex v of minimum degree, remove v from the graph, color the resulting graph recursively, then insert v back into the graph and assign it the smallest available color. At the time v is removed (and thus also at the time it is inserted back again), the degree $d(v)$ of v is at most the degree $d(u)$ of the vertex u corresponding to a disk with the smallest y-coordinate. Thus, v is assigned color at most $d(v)+1$, while the optimal coloring needs at least $(d(u)+1)/3 \geq (d(v)+1)/3$ colors. This shows that approximation ratio 3 is achieved.

Theorem 10 (Peeters [24], Marathe et al. [22]). *There is a* 3*-approximation algorithm for coloring unit disk graphs that does not require the disk representation.*

One can even show that the algorithm uses at most $3\omega(G) - 2$ colors, where $\omega(G)$ is the size of a maximum clique in the given unit disk graph. We sketch the idea for proving this in the case of given disk representation. When the algorithm colors a disk D, all previously colored neighbors have a y-coordinate not smaller than D. Thus, the previously colored neighbors can be covered by at most three cliques (see Fig. 1 a)): The disks with centers in the same region delimited by an angle of $\frac{\pi}{3}$ must form a clique (together with D). Thus, $d(D)$ is at most $3\omega(G) - 3$, so the algorithm uses at most $3\omega(G) - 2$ colors. In the case that the representation is not given, the smallest-degree-last First-fit algorithm uses at most $3\omega(G) - 2$ colors on unit disk graphs as well.

3.2 Off-line coloring of general disk graphs

The approach for unit disk graphs was generalized to disk graphs by Marathe et al. [22]. They proved that the achieved approximation ratio for coloring disk graphs is at most 6, but their analysis can be sharpened to show that the smallest-degree-last First-fit algorithm is in fact a 5-approximation algorithm (cf. Gräf [9] and Malesińska [21]). To see this, note that the closed neighborhood of a disk with smallest diameter cannot contain an independent set of size 6 and hence such a disk has degree at most $5(\chi(G) - 1)$, where $\chi(G)$ is the optimal number of colors (i.e., the chromatic number). Consequently, if a disk with the smallest degree is colored last, this disk has at most $5(\chi(G)-1)$ previously colored neighbors and is assigned color at most $5\chi(G) - 4$. Since the graph obtained after removing the disk of smallest degree is again a disk graph, the argument can be applied recursively. Thus, approximation ratio 5 is achieved. If

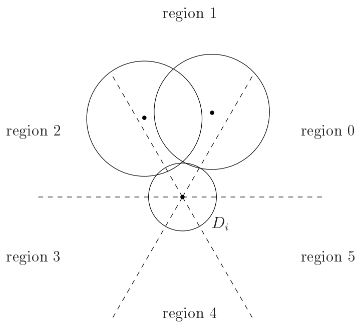

Fig. 4. The larger disks intersecting D_i can be partitioned into six cliques.

the disk representation is given, we can alternatively sort the disks by non-increasing diameter and apply First-fit in this order to achieve the same bound.

Theorem 11. *The smallest-degree-last First-fit algorithm achieves approximation ratio at most* 5 *for disk graphs. It does not need the disk representation.*

Furthermore, it can be shown that every disk graph G can be colored with at most $6\omega(G) - 6$ colors (Gräf [9] and Malesińska [21]). Consider the smallest disk D_i and the disks that intersect it. See Fig. 4. The plane around D_i can be partitioned into six regions delimited by an angle of $\frac{\pi}{3}$. Since all neighboring disks are at least as big as D_i, simple geometric arguments show that the neighboring disks whose centers are in the same region must form a clique. Furthermore, the regions can be chosen such that the center of at least one neighboring disk lies on the border between two regions, so that this neighboring disk belongs to the cliques of both regions. Therefore, the neighborhood of D_i can contain at most $6(\omega(G) - 1) - 1$ disks. Consequently, the algorithm that colors the smallest disk last needs at most $6\omega(G) - 6$ colors.

So far the only lower bound on the approximability of disk graph coloring is the one derived from the *NP*-completeness of deciding 3-colorability. It would be interesting to derive a larger lower bound that uses specific properties of disk graphs.

For σ-bounded disk graphs, no better approximation algorithms than for general disk graphs have been presented in the literature, although we expect that the upper bound on the approximation ratio achieved by the smallest-degree-last First-fit algorithm drops from 5 to 4 on disk graphs whose diameter ratio is bounded by a value σ that is smaller than some threshold (but still greater than 1).

On-line coloring of disk graphs and σ-bounded disk graphs is treated in detail in the next section.

4 New results for on-line coloring of disk graphs

As a motivation for our further study of on-line coloring of disk graphs we shall mention the result of Gyárfás and Lehel from 1988 showing that there exists a tree T on n vertices such that for every on-line coloring algorithm there exists a specific ordering of the vertices of T, such that the algorithm is forced to use $\Omega(\log n)$ distinct colors [11]. Every tree is planar, so this result together with the theorem of Koebe immediately shows that this lower bound is valid also for disk graphs *without given representation*. In the following we will prove that even the knowledge of the disk representation does not admit a better performance of a coloring algorithm. We prove that for every on-line disk coloring algorithm there exists a sequence of n disks such that the algorithm is forced to use at least $\Omega(\log n)$ distinct colors, while an optimal coloring uses only two colors. We also adapt a result of Irani [16] and show that a competitive ratio of $O(\log n)$ is achieved by the First-fit coloring algorithm. This shows that the First-fit algorithm is optimal for on-line coloring of disk graphs up to a constant factor.

For simplicity, we first prove the lower bound result for intersection graphs of squares (Subsection 4.1) and then provide arguments how the proof can be adapted to disks (Subsection 4.2). In Subsection 4.3, we prove that the First-fit heuristic gives competitive ratio $O(\log n)$ for disk graphs and square graphs. In Subsection 4.4, we consider σ-bounded disk graphs. For the case with given representation, we use a different algorithm that achieves an improved competitive ratio of $O(\min\{\log n, \log \sigma\})$. For the case without representation, we get an upper bound of $O(\min\{\log n, \sigma^2\})$ on the competitive ratio of the First-fit algorithm.

4.1 A lower bound for on-line coloring of squares

Let $\mathcal{A}$ be an arbitrary on-line square coloring algorithm. We prove that for any n there exists a sequence of squares $S_1, S_2, ..., S_m = \mathcal{S}(\mathcal{A}, n)$, (where $m = m(\mathcal{A}, n)$ depends on $\mathcal{A}$ and n, and is bounded by 2^n), such that

- The intersection graph of $\mathcal{S}(\mathcal{A}, n)$ is isomorphic to a tree.
- The algorithm $\mathcal{A}$ is forced to use at least n distinct colors.

Let us start with some auxiliary notions:

We deal only with squares whose sides are parallel to the axes of the coordinate system. Each square S_i is uniquely delimited by bounds $(\underline{x_i}, \underline{y_i}, \overline{x_i}, \overline{y_i})$, satisfying $\overline{x_i} - \underline{x_i} = \overline{y_i} - \underline{y_i}$. In particular, the symbol $\square$ stands for the unit square which is delimited by $(0,0,1,1)$.

For a square S the operation $S \cdot c + (x,y)$ means the linear transformation where all coordinates of S are first multiplied by the constant c and then shifted by the vector (x,y).

If $\mathcal{S}$ and $\mathcal{S}'$ are sequences of squares, then their concatenation is denoted by $\mathcal{S} \circ \mathcal{S}'$. If $\mathcal{S}$ is a nonempty sequence of squares, then $\mathcal{S}\flat$ is the same sequence without the last element.

We say that squares of some sequence are in general position if every pair of squares in the set differs in the $\overline{y}$ coordinate.

Now assume that the intersection graph of an arbitrary sequence of squares $S_1, ..., S_k$ in general position is a forest F. In each connected component of F we define the *active square* to be the one with the highest $\overline{y}$ coordinate. The *active zone* of an active square S_i is delimited by interval $\langle y_\circ, y^\circ \rangle$ where $y^\circ = \overline{y_i}$ and $y_\circ = \max_j\{\overline{y_j} : S_j \neq S_i$ and S_j belongs to the same connected component as $S_i\}$. If the connected component of S_i contains only S_i, we let $y_\circ = \underline{y_i}$.

The active zone of a sequence $\mathcal{S} = \{S_1, ..., S_k\}$ is defined as the intersection of the active zones of all active squares in $\mathcal{S}$. The width of an active zone is equal to the length of the corresponding interval, or is equal to 0 if the zone is empty.

Our construction of $S_1, ..., S_m = \mathcal{S}(\mathcal{A}, n)$ satisfies the following invariant:

Invariant 1 *For each on-line algorithm $\mathcal{A}$ and each $n \geq 2$, there exists a sequence $\mathcal{S}(\mathcal{A}, n)$ of at most 2^n squares such that*

- *for every $S \in \mathcal{S}(\mathcal{A}, n)$, we have $S \subseteq \square$,*
- *the squares in $\mathcal{S}$ are in general position,*
- *the intersection graph of $\mathcal{S}(\mathcal{A}, n)$ is a tree,*
- *the active zones of $\mathcal{S}(\mathcal{A}, n)$ and $\mathcal{S}(\mathcal{A}, n)\flat$ have positive width,*
- *on the active squares of $\mathcal{S}(\mathcal{A}, n)\flat$ the algorithm $\mathcal{A}$ uses at least $n-1$ distinct colors, and*
- *the last square in $\mathcal{S}(\mathcal{A}, n)$ intersects all active squares of $\mathcal{S}(\mathcal{A}, n)\flat$ and is the active square of $\mathcal{S}(\mathcal{A}, n)$.*

We present a construction of $\mathcal{S}(\mathcal{A}, n)$ satisfying the invariant by induction.

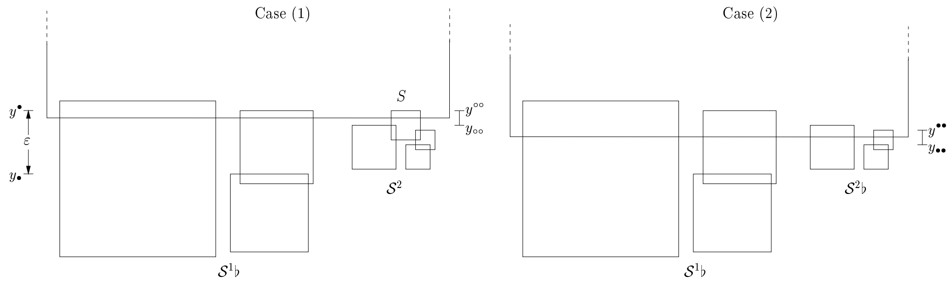

Fig. 5. Illustration of lower bound construction.

Our statement is clearly true for $n = 2$, when we select $\mathcal{S}(\mathcal{A}, 2)$ as two intersecting squares, e.g. delimited by $(\frac{1}{5}, \frac{1}{5}, \frac{3}{5}, \frac{3}{5})$ and $(\frac{2}{5}, \frac{2}{5}, \frac{4}{5}, \frac{4}{5})$.

Now assume that the hypothesis is valid for n and we want to prove the statement for $n + 1$.

Let $\mathcal{B}$ be the algorithm derived from the algorithm $\mathcal{A}$ which gives to a square $S_i \subseteq \square$ the same color as $\mathcal{A}$ gives to $S_i \cdot \frac{1}{4}$.

By the induction hypothesis there exists a set $\mathcal{S}(\mathcal{B}, n)$, such that $\mathcal{B}$ on $\mathcal{S}(\mathcal{B}, n)\flat$ uses at least $n - 1$ distinct colors, or equivalently there exists a set $\mathcal{S}^1$ satisfying the invariant for $\mathcal{A}$ and n, and moreover all squares of $\mathcal{S}^1$ are inside $\square \cdot \frac{1}{4}$.

Assume that the active zone of $\mathcal{S}^1\flat$ forms an interval $\langle y_\bullet, y^\bullet \rangle$ and let ε be the width of this interval, i.e. $\varepsilon = y^\bullet - y_\bullet \leq \frac{1}{4}$.

Denote by $\mathcal{B}'$ the algorithm which behaves exactly as $\mathcal{A}$ after it colors $\mathcal{S}^1\flat$, and which gives to a box $S_i \subseteq \square$ the same color as $\mathcal{A}$ gives to $S_i \cdot \varepsilon + (\frac{5}{12}, y_\bullet)$.

Now by the hypothesis on $\mathcal{B}'$, there exists a sequence $\mathcal{S}(\mathcal{B}', n)$ such that $\mathcal{B}'$ uses on active squares of $\mathcal{S}(\mathcal{B}', n)\flat$ at least $n - 1$ distinct colors, and as above this implies the existence of a set $\mathcal{S}^2$ satisfying the invariant for $\mathcal{A}$ and n, and moreover all squares of $\mathcal{S}^2$ are inside the active zone of $\mathcal{S}^1\flat$, but disjoint from $\mathcal{S}^1$.

Now we have to consider two cases (illustrated for $\mathcal{S}(\mathcal{A}, 4)$ in Fig. 5):

(1) The sets of colors used on active squares of $\mathcal{S}^1\flat$ and $\mathcal{S}^2\flat$ are the same. The last square S of $\mathcal{S}^2$ uses the new n-th color. Let the active zone of S be $\langle y_{\circ\circ}, y^{\circ\circ} \rangle$. We put

$$\mathcal{S}(\mathcal{A}, n+1) = \mathcal{S}^1\flat \;\circ\; \mathcal{S}^2 \;\circ\; \left\{\square \cdot \frac{2}{3} + \left(0, \frac{y_{\circ\circ} + y^{\circ\circ}}{2}\right)\right\},$$

$m(\mathcal{A}, n+1) = m(\mathcal{B}, n) + m(\mathcal{B}', n)$, and the hypothesis is satisfied.

(2) Active squares of $\mathcal{S}^1\flat$ and $\mathcal{S}^2\flat$ have different colors. This means that there appear at least n different colors. Observe that the active zone $\langle y_{\bullet\bullet}, y^{\bullet\bullet}\rangle$ of $\mathcal{S}^1\flat \circ \mathcal{S}^2\flat$ has positive width. Then we put

$$\mathcal{S}(\mathcal{A}, n+1) = \mathcal{S}^1\flat \;\circ\; \mathcal{S}^2\flat \;\circ\; \left\{\square \cdot \frac{2}{3} + \left(0, \frac{y_{\bullet\bullet} + y^{\bullet\bullet}}{2}\right)\right\},$$

$m(\mathcal{A}, n+1) = m(\mathcal{B}, n) + m(\mathcal{B}', n) - 1$, and the invariant is again valid.

Observe that our construction satisfies:

- all squares are in general position,
- dimensions are selected such that all squares are placed inside the unit square $\square$, and
- the last square of $\mathcal{S}(\mathcal{A}, n+1)$ intersects all active squares (and no other squares), hence the intersection graph is a tree.

Therefore, the construction constitutes a proof for the following theorem.

Theorem 12. *For every on-line square coloring algorithm $\mathcal{A}$ there exists a sequence of n squares $\mathcal{S}$ such that $\mathcal{A}$ uses $\Omega(\log n)$ distinct colors on $\mathcal{S}$ while $\mathcal{S}$ can be colored optimally with two colors.*

4.2 A lower bound for on-line coloring of disks

Instead of giving a detailed proof of the lower bound for disk graphs, we show how to adapt the approach of the previous section for square graphs and only point out which aspects require a different treatment.

We define the active zone of an active disk similarly, i.e. as an interval $\langle y_\circ, y^\circ\rangle$, where y° is the maximum y-coordinate of the active disk and $y_\circ$ is the maximum y-coordinate of any other disk in the same connected component. If the active disk does not intersect any other disk, we take $y_\circ$ to be its minimum y-coordinate. We adapt the construction of the previous section in order to satisfy the following invariant:

Invariant 2 *For each on-line coloring algorithm $\mathcal{A}$ and each $n \geq 2$, there exists a sequence of at most 2^n disks $\mathcal{D}(\mathcal{A}, n)$ such that*

- *for every $D \in \mathcal{D}(\mathcal{A}, n)$, we have $D \subseteq \square$,*
- *the disks in $\mathcal{D}$ are in general position,*
- *the intersection graph of $\mathcal{D}(\mathcal{A}, n)$ is a tree,*
- *the active zones of $\mathcal{D}(\mathcal{A}, n)$ and $\mathcal{D}(\mathcal{A}, n)\flat$ have positive width,*

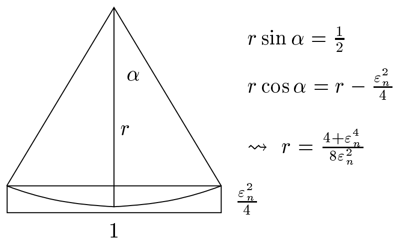

Fig. 6. Disk intersecting an active zone of width $\varepsilon_n^2/4$.

- *on the active disks of $\mathcal{D}(\mathcal{A}, n)\flat$ the algorithm $\mathcal{A}$ uses at least $n-1$ distinct colors, and*
- *the last disk in $\mathcal{D}(\mathcal{A}, n)$ intersects the active zones of all active disks of $\mathcal{D}(\mathcal{A}, n)\flat$ and is the active disk of $\mathcal{D}(\mathcal{A}, n)$.*

The only problem that might arise in the new construction is that the diameter of the last disk D_m in $\mathcal{D}(\mathcal{A}, n+1)$ might have to be very large in order to ensure that this disk intersects the tiny active zones of all active disks in $\mathcal{D}(\mathcal{A}, n+1)\flat$, but no other disks. In this case, the first condition of the invariant might be violated. However, we can handle this issue as follows: We evaluate all finitely many possible cases that might arise in the construction of $\mathcal{D}(\mathcal{A}, n+1)$ and determine the maximum diameter Δ that is required for the disk D_m. Then we select the magnification factor for the algorithms $\mathcal{B}$ and $\mathcal{B}'$ depending on Δ such that the last active disk remains inside the unit square $\square$ in any case that may arise after the recursive construction of the sets $\mathcal{D}(\mathcal{B}, n)$ and $\mathcal{D}(\mathcal{B}', n)$.

More precisely, for any n let ε_n be the minimum width of the active zones of $\mathcal{D}(\mathcal{A}, n)$ and $\mathcal{D}(\mathcal{A}, n)\flat$, taken over all on-line algorithms $\mathcal{A}$. Since we have shown that the length of $\mathcal{D}(\mathcal{A}, n)$ is at most 2^n, and since our construction of $\mathcal{D}(\mathcal{A}, n)$ has always at most two choices for the next disk (corresponding to the cases (1) and (2) in Subsection 4.1), there are only finitely many possible sets $\mathcal{D}(\mathcal{A}, n)$, and the class of all on-line algorithms can be partitioned into a finite number of equivalence classes with respect to the active zones of $\mathcal{D}(\mathcal{A}, n)$ and $\mathcal{D}(\mathcal{A}, n)\flat$. Therefore, ε_n is well-defined.

Following the notation of the proof of Invariant 1 (but writing $\mathcal{D}^1$ and $\mathcal{D}^2$ in place of $\mathcal{S}^1$ and $\mathcal{S}^2$), we may assume for algorithm $\mathcal{A}$ by the inductive hypothesis that the width of the active zone of $\mathcal{D}^1\flat$ is at least $\varepsilon_n \cdot \frac{1}{4}$ and similarly the width of the active zones of $\mathcal{D}^2$ and $\mathcal{D}^2\flat$ is at least $\varepsilon_n^2 \cdot \frac{1}{4}$. A final disk intersecting these active zones (represented as a rectangle of height $\varepsilon_n^2 \cdot \frac{1}{4}$ and unit width in Fig. 6) can be found with radius $\frac{5}{8\varepsilon_n^2}$. So if we use an additional scaling factor equal to $\frac{1}{2}\varepsilon_n^2$, we ensure that all disks are placed inside the unit square $\square$. In addition, we see that the minimum width of the active zones of $\mathcal{D}(\mathcal{A}, n+1)$ and $\mathcal{D}(\mathcal{A}, n+1)\flat$ is at least $\varepsilon_{n+1} \geq \frac{1}{8}\varepsilon_n^4$. As we can choose $\varepsilon_2 = \frac{1}{5}$, we get that $\varepsilon_n \geq 2 \cdot 10^{-4^{n-2}}$ for all $n \geq 2$.

Theorem 13. *For every on-line disk coloring algorithm $\mathcal{A}$ there exists a sequence of n disks $\mathcal{D}$ such that $\mathcal{A}$ uses at least $\Omega(\log n)$ distinct colors on $\mathcal{D}$ while $\mathcal{D}$ can be colored optimally with two colors.*

Furthermore, by analyzing the diameter of the smallest disk used in the construction of $\mathcal{D}(\mathcal{A}, n)$, we see that the diameter-ratio σ of $\mathcal{D}(\mathcal{A}, n)$ is bounded from above by $O(10^{4^{n-2}})$. Since $\mathcal{A}$ uses at least n colors on this instance, the competitive ratio of $\mathcal{A}$ must be at least $\Omega(\log\log\sigma)$.

Theorem 14. *No deterministic on-line disk coloring algorithm can have competitive ratio $o(\log\log\sigma)$ on σ-bounded disk graphs.*

4.3 Upper bound for the First-fit algorithm

In order to analyze the competitive ratio of First-fit for on-line coloring of squares or disks, we make use of a result due to Irani [16]. A graph G is called *d-inductive* if the vertices of G can be ordered in such a way that each vertex has at most d edges to higher-numbered vertices. Irani proved that if G is a d-inductive graph on n nodes, then First-fit uses $O(d\log n)$ colors to color G. In order to apply this result to disk graphs and square graphs, we will show that these graphs are $O(\omega(G))$-inductive, where $\omega(G)$ denotes the size of a maximum clique in a graph G. We reformulate Irani's result in terms of the competitive ratio of First-fit on $O(\omega(G))$-inductive graphs.

Theorem 15 (Irani [16]). *Let d be a constant and let $\mathcal{G}$ be a class of graphs such that every $G \in \mathcal{G}$ is $d\omega(G)$-inductive. Then the First-fit coloring algorithm is $O(d\log n)$-competitive on graphs from $\mathcal{G}$, where n is the number of vertices of the given graph.*

Proof. We first introduce some terminology used later in the proof. Let G be a graph from the class $\mathcal{G}$. Let $v_1, v_2, \ldots, v_n$ be the order in which the vertices of G are colored. We write ω instead of $\omega(G)$. View each edge of G as being directed from the endpoint with smaller number to the endpoint with higher number in some vertex ordering that witnesses that G is $d\omega$-inductive. The fact that a directed edge goes from u to v is written as $u \succ v$. Then $\operatorname{outdeg}(u) = |\{v : u \succ v\}|$. Note that $\operatorname{outdeg}(u) \leq d\omega$

since G is $d\omega$-inductive. For a vertex v_j we define its set of successors as $S_j = \{v_i : v_j \succ v_i,\ i > j\}$. Observe that $|S_j| \le d\omega$.

Assign to every vertex $v_j \in V(G)$ the value $C_j = 1$ and perform the following discharging procedure:

```
DISCHARGING PROCEDURE
for j := 1 to n do
  begin
    W_j := C_j;
    if S_j ≠ ∅ then
      for all v_i ∈ S_j do C_i := C_i + C_j/|S_j|
    C_j := 0;
  end.
```

In every round of the outer for loop we get that

$$1 \le W_j = C_j \le \sum_{j=1}^{n} C_j \le n.$$

We now bound W_j in terms of the maximum assigned color. Assume that a vertex v_j gets color c by the First-fit algorithm.

We prove by induction on c that $\frac{W_j}{|S_j|} \ge \frac{1}{d\omega}(1+\frac{1}{d\omega})^{c-1}$ whenever $S_j \ne \emptyset$, and $W_j \ge (1+\frac{1}{d\omega})^{c-1-d\omega}$ otherwise.

The statement is true for $c = 1$. For $c > 1$, there must be $c-1$ vertices v_i with $i < j$ that are adjacent to v_j and that have been assigned different colors. At most $d\omega - |S_j|$ of these $c-1$ vertices can come after v_j in the inductive order, so there must be at least $c - 1 - (d\omega - |S_j|)$ predecessors of v_j: $P_j = \{v_i : v_i \succ v_j,\ j > i\}$ that have been assigned different colors. Then, by the induction hypothesis:

$$\frac{W_j}{|S_j|} \ge \frac{1}{|S_j|}\left[1 + \sum_{v_i \in P_j} \frac{W_i}{|S_i|}\right] \ge \frac{1}{|S_j|}\left[1 + \frac{1}{d\omega} \sum_{i=1}^{c-1-d\omega+|S_j|} \left(1+\frac{1}{d\omega}\right)^{i-1}\right] \ge$$
$$\ge \frac{1}{|S_j|}\left(1+\frac{1}{d\omega}\right)^{c-1-d\omega+|S_j|}.$$

The last term is minimized in the case $|S_j| = d\omega$ and the claim follows. The case $S_j = \emptyset$ is discussed similarly.

We get that $n \ge \frac{1}{d\omega}(1+\frac{1}{d\omega})^{c-1-d\omega}$ and therefore $c = O(d\omega \log n)$. $\square$

Now we give bounds on the inductiveness of disk graphs and square graphs in terms of $\omega(G)$.

Lemma 1. *Every disk graph G is $6\omega(G)$-inductive.*

Proof. Let $\mathcal{D}$ be a set of disks that is a disk representation of G. Order the disks in $\mathcal{D}$ according to non-decreasing diameters. Consider some disk D_i. By the arguments of the discussion after Theorem 11, the higher-numbered disks that intersect D_i can be partitioned into six groups such that the disks in each group form a clique. $\square$

Lemma 2. *Every square graph G is $4\omega(G)$-inductive.*

Proof. Order the squares in order of non-decreasing size. Consider some square S_i. All higher-numbered squares that intersect S_i contain at least one corner of S_i. Therefore, the higher-numbered squares that intersect S_i can be partitioned into four cliques. $\square$

From Theorem 15 and Lemmas 1 and 2 we get the following result.

Theorem 16. *First-fit uses $O(\omega(G)\log n)$ colors to color a disk graph or square graph G with n nodes and is thus an $O(\log n)$-competitive on-line algorithm. It does not require the geometric representation.*

4.4 Coloring disks with bounded diameter ratio

We now focus our attention on the case that the ratio of the diameter of the largest disk and the smallest one is bounded by some value σ and that the disk representation is given as part of the input. (In fact, it would suffice that the diameters of the disks are given as part of the input.)

We prove that there exists an on-line coloring algorithm with competitive ratio $O(\min\{\log n, \log\sigma\})$. The algorithm is a composition of two methods: The first method $\mathcal{A}$ is the First-fit technique for disk graphs with arbitrary diameter. It provides the bound $O(\log n)$. The second method $\mathcal{B}$ is the First-fit method applied separately on $\log\sigma$ layers of disks, where the diameters of the disks on each layer are within a factor of two so that First-fit has constant competitive ratio on each layer.

More precisely, the algorithm $\mathcal{B}$ colors disks $D_1, \ldots, D_n$ as follows:

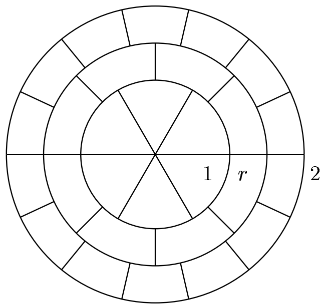

Fig. 7. 28 segments of the plane around the center of D_i, $r = 1.306$.

```
FIRST-FIT ON LAYERS B
L_j := ∅ for all j ∈ Z;
for i := 1 to n do
  begin
    j := ⌊log_2(diam(D_i))⌋;
    L_j := L_j ∪ {D_i}; % the layer containing D_i %
    F := {c(D_k) : D_k ∈ L_j, D_k ∩ D_i ≠ ∅}∪
      {c(D_k) : 1 ≤ k ≤ i, D_k ∉ L_j}; % the set of forbidden colors %
    c(D_j) := min(N \ F)
  end.
```

Lemma 3. *The First-fit coloring algorithm is 28-competitive on disks of diameter ratio bounded by two.*

Proof. Assume that disks in the set $\mathcal{D}$ are scaled such that the smallest disk has unit diameter. The centers of all disks that intersect a particular disk D_i are at distance at most two from the center of D_i. We divide the plane around the center of D_i up to distance two into 28 segments such that inside each segment, every pair of points is at distance at most one, see Fig. 7. If the centers of two disks lie in the same segment, then these disks intersect, and hence disks with centers in the same segment induce a clique in the intersection graph G of $\mathcal{D}$.

Further we observe that the six inner segments contain also the center of D_i and therefore the vertex D_i can have at most $28\omega(G) - 6$ neighbors in G. Then the First-fit coloring algorithm uses at most $28\omega(G) - 5$ colors on G, and hence is 28-competitive. □

Since algorithm $\mathcal{B}$ achieves constant competitive ratio on each layer and there are only $O(\log \sigma)$ different layers, we obtain the following lemma.

Lemma 4. *If the disk representation is given as part of the input, the First-fit method on layers is an $O(\log \sigma)$-competitive coloring algorithm for disk graphs of diameter ratio bounded by σ.*

We combine First-fit and First-fit on layers as follows: We use two separate sets of colors for the algorithms $\mathcal{A}$ and $\mathcal{B}$. When a new disk D_i is presented we run $\mathcal{A}$ on D_i together with those disks colored by $\mathcal{A}$. Similarly we execute $\mathcal{B}$. Then we compare the results of these two algorithms and color D_i with the algorithm that has used fewer colors up to now (including disk D_i). The total number of colors used on the entire set $\mathcal{D}$ is the sum of the number of colors used by both methods. Note that at any time of the execution of the combined algorithm, the number of colors used by $\mathcal{A}$ and the number of colors used by $\mathcal{B}$ differ by at most one.

Assume that $\log n < \log \sigma$. The number of colors used by algorithm $\mathcal{A}$ is at most $O(\log n)$ times the optimal number. The number of colors used by algorithm $\mathcal{B}$ is at most one more than that of $\mathcal{A}$. So the total number of colors used by the combined algorithm is $O(\log n)$ times the optimal number of colors. A symmetric argument holds in the case that $\log \sigma \leq \log n$. Therefore, we obtain the following theorem.

Theorem 17. *If the disk representation is given as part of the input, there is an $O(\min\{\log n, \log \sigma\})$-competitive coloring algorithm for disk graphs whose diameter ratio is bounded by σ.*

We remark that an analogous result can be obtained for square intersection graphs as well.

Concerning on-line coloring of σ-bounded disk graphs in the case without given disk representation, First-fit is easily seen to be $O(\sigma^2)$-competitive. This follows because the neighborhood of a disk can be covered by $O(\sigma^2)$ cliques. The idea of the analysis is the same as the one used in the proof of Lemma 3 for the case $\sigma = 2$. Combining this with the upper bound of $O(\log n)$, we get that First-fit is $O(\min\{\log n, \sigma^2\})$-competitive on σ-bounded disk graphs.

5 Conclusion

We have given a survey of known upper and lower bounds on the approximation ratio and competitive ratio achievable for the maximum independent set and minimum coloring problems on disk graphs. The bounds are summarized in Table 1.

Problem	Graph class	Disk rep.	Approximation ratio lower	Approximation ratio upper	Competitive ratio lower	Competitive ratio upper
Maximum independent set	UDG	+	NPC	PTAS	5	
	DG$_\sigma$	+	NPC	PTAS	$\Theta(\min\{n, \sigma^2\})$	
	DG	+	NPC	PTAS	$n-1$	
	UDG	−	NPC	3	5	
	DG$_\sigma$	−	NPC	5	$\Theta(\min\{n, \sigma^2\})$	
	DG	−	NPC	5	$n-1$	
Minimum coloring	UDG	+	4/3	3	2	5
	DG$_\sigma$	+	4/3	5	$\Omega(\log\log\sigma)$	$O(\min\{\log n, \log\sigma\})$
	DG	+	4/3	5	$\Omega(\log n)$	$O(\log n)$
	UDG	−	4/3	3	2	5
	DG$_\sigma$	−	4/3	5	$\Omega(\log\log\sigma)$	$O(\min\{\log n, \sigma^2\})$
	DG	−	4/3	5	$\Omega(\log n)$	$O(\log n)$

Table 1. Summary of known results for disk graphs with n vertices. UDG stands for unit disk graphs, DG$_\sigma$ for σ-bounded disk graphs, and DG for general disk graphs. PTAS stands for polynomial-time approximation scheme, NPC for *NP*-complete.

For on-line coloring of disk graphs, we have shown that the First-fit method, which does not need the disk representation as part of the input, provides an $O(\log n)$-competitive disk coloring algorithm and that no algorithm can have competitive ratio $o(\log n)$, even if it uses the geometric representation. Furthermore, we showed that the geometric representation can help to get a better ratio of $O(\min\{\log n, \log\sigma\})$ on instances with diameter-ratio bounded by σ. Analogous results hold for intersection graphs of squares.

We initiated here the study of on-line coloring of disk graphs with diameter ratio bounded by σ. However, for this particular problem the gap between the lower bound $\Omega(\log\log\sigma)$ and the upper bound $O(\log\sigma)$ on the competitive ratio should be narrowed. Furthermore, we believe that the use of methods like randomization might improve the competitive ratio for the on-line problems studied here, and we expect further results in this direction.

The most widely used lower bound on the chromatic number of a disk graph (i.e., the lower bound on the optimal solution) is expressed via the clique number of the graph. We hope that by use of more sophisticated arguments it could be proven that standard coloring algorithms behave even better. As a particular open problem we would ask what is the supremum of the ratio of chromatic and clique number of a unit disk graph. The only known bounds are $1.5 \leq \frac{\chi(G)}{\omega(G)} < 3$. The lower bound is derived from the coloring of the cycle C_5, and the upper bound is achieved by the algorithm due to Peeters [24].

It should be mentioned that approximation algorithms for other *NP*-hard optimization problems besides maximum independent set and minimum coloring have also been studied for unit disk graphs and disk graphs. These problems include the weighted version of the maximum independent set problem, the minimum (weight) vertex cover problem, and the minimum dominating set problem (and variants thereof) [22, 15, 21, 23, 7]. Many of these results can be adapted to intersection graphs of arbitrary regular polygons (including squares), also in higher dimensions. One of the interesting open problems is to determine whether a polynomial-time approximation scheme exists for the minimum dominating set problem in general disk graphs with given representation. Another open problem is to settle the complexity of the maximum clique problem for general disk graphs (the problem is polynomial for unit disk graphs [6, 25]).

Furthermore, a polynomial-time approximation scheme for the weighted fractional coloring problem on disk graphs was obtained by Jansen and Porkolab [17, 18]. Previously, a 2-approximation algorithm for fractional coloring of unit disk graphs had been presented by Matsui [23].

Finally, we would like to point out that many questions are still open for rectangle intersection graphs, a generalization of square intersection graphs. The recognition problem for rectangle intersection graphs has been proved *NP*-complete by Kratochvíl [20]. For the maximum independent set problem in rectangle intersection graphs with n vertices, an $O(\log n)$-approximation algorithm was presented for the case of given geometric representation [1], but it is unknown whether a better approximation ratio can be achieved in general. For coloring a rectangle intersection graph G with clique number $\omega(G)$, it is known that $O(\omega(G)^2)$ colors suffice [5], but no example with a non-linear lower bound in terms of the clique number has been obtained. For some special cases, it is known that $O(\omega(G))$ colors suffice [21]. It would be an interesting problem for future research to further investigate off-line and on-line coloring of rectangle intersection graphs.

References

1. P. K. Agarwal, M. van Kreveld, and S. Suri. Label placement by maximum independent set in rectangles. *Computational Geometry: Theory and Applications*, 11(3–4):209–218, 1998.
2. B. Baker. Approximation algorithms for NP-complete problems on planar graphs. *Journal of the ACM*, 41(1):153–180, 1994.

3. H. Breu and D. G. Kirkpatrick. On the complexity of recognizing intersection and touching graphs of disks. In *Proceedings of the Symposium on Graph Drawing (GD'95)*, Lecture Notes in Computer Science 1027, pages 88–98, 1996.
4. H. Breu and D. G. Kirkpatrick. Unit disk graph recognition is NP-hard. *Computational Geometry: Theory and Applications*, 9(1–2):3–24, 1998.
5. J. Burling. *On coloring problems of families of prototypes.* PhD thesis, University of Colorado, 1965.
6. B. N. Clark, C. J. Colbourn, and D. S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86(1–3):165–177, 1990.
7. T. Erlebach, K. Jansen, and E. Seidel. Polynomial-time approximation schemes for geometric graphs. In *Proceedings of the 12th Annual ACM–SIAM Symposium on Discrete Algorithms SODA'01*, pages 671–679, 2001.
8. J. Fiala, A. V. Fishkin, and F. V. Fomin. Off-line and on-line distance constrained labeling of graphs. In *Proceedings of the 9th European Symposim on Algorithms ESA'01*, Lecture Notes in Computer Science 2161, pages 464–475. Springer Verlag, 2001.
9. A. Gräf. Coloring and recognizing special graph classes. Technical Report Musikinformatik und Medientechnik Bericht 20/95, Johannes Gutenberg-Universität Mainz, 1995.
10. A. Gräf, M. Stumpf, and G. Weißenfels. On coloring unit disk graphs. Technical Report Musikinformatik und Medientechnik Bericht 17/94, Johannes Gutenberg-Universität Mainz, 1994.
11. A. Gyárfás and J. Lehel. On-line and first fit colourings of graphs. *Jornal of Graph Theory*, 12(2):217–227, 1988.
12. W. K. Hale. Frequency assignment: Theory and applications. *Proc. of the IEEE*, 68(12):1497–1514, 1980.
13. P. Hliněný and J. Kratochvíl. Representing graphs by disks and balls. *Discrete Mathematics*, 229(1–3):101–124, 2001.
14. D. Hochbaum and W. Maass. Approximation schemes for covering and packing problems in image processing and VLSI. *Journal of the ACM*, 32(1):130–136, 1985.
15. H. B. Hunt III, M. V. Marathe, V. Radhakrishnan, S. S. Ravi, D. J. Rosenkrantz, and R. E. Stearns. NC-Approximation schemes for NP- and PSPACE-hard problems for geometric graphs. *Journal of Algorithms*, 26(2):238–274, 1998.
16. S. Irani. Coloring inductive graphs on-line. *Algorithmica*, 11:53–72, 1994.
17. K. Jansen. Approximate strong separation with application in fractional graph coloring and preemptive scheduling. Manuscript, 2001.
18. K. Jansen and L. Porkolab. On preemptive resource constrained scheduling: Polynomial-time approximation schemes. Manuscript, 2001.
19. P. Koebe. Kontaktprobleme der konformen Abbildung. *Ber. Verh. Sächs. Akad. Leipzig*, 88:141–164, 1936.
20. J. Kratochvíl. A special planar satisfiability problem and a consequence of its NP-completeness. *Discrete Applied Mathematics*, 52:233–252, 1994.
21. E. Malesińska. *Graph theoretical models for frequency assignment problems.* PhD thesis, Technical University of Berlin, 1997.
22. M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi, and D. J. Rosenkrantz. Simple heuristics for unit disk graphs. *Networks*, 25:59–68, 1995.
23. T. Matsui. Approximation algorithms for maximum independent set problems and fractional coloring problems on unit disk graphs. In *Discrete and Computational Geometry*, Lecture Notes in Computer Science 1763, pages 194–200, 2000.

24. R. Peeters. On coloring j-unit sphere graphs. Technical report, Dept. of Economics, Tilburg University, 1991.
25. V. Raghavan and J. Spinrad. Robust algorithms for restricted domains. In *Proceedings of the 12th Annual ACM–SIAM Symposium on Discrete Algorithms SODA'01*, pages 460–467, 2001.
26. H. Sachs. Coin graphs, polyhedra, and conformal mapping. *Discrete Mathematics*, 134:133–138, 1994.
27. D. Wang and Y.-S. Kuo. A study on two geometric location problems. *Information Processing Letters*, 28:281–286, 1988.
28. A. Wolff and T. Strijk. The map-labeling bibliography. http://www.math-inf.uni-greifswald.de/map-labeling/bibliography/.