Fine-grained complexity of coloring unit disks and balls

Csaba Biró*, Édouard Bonnet†, Dániel Marx†, Tillmann Miltzow†, Paweł Rzążewski†‡

Abstract

We investigate the possible complexity of an algorithm deciding the ℓ-colorability of an intersection graph of unit disks. We exhibit a smooth increase of complexity as the number ℓ of colors increases: If we restrict the number of colors to $\ell = \Theta(n^\alpha)$ for some $0 \leq \alpha \leq 1$, then the problem of coloring the intersection graph of n unit disks with ℓ colors

- can be solved in time $\exp\Big(O(\sqrt{n\ell}\log n)\Big)$, and
- cannot be solved in time $\exp\Big(o(\sqrt{n\ell})\Big)$, unless the ETH fails.

More generally, we consider the problem of coloring d-dimensional unit balls in the Euclidean space and obtain analogous results showing that the problem

- can be solved in time $\exp\big(O(n^{1-1/d}\ell^{1/d}\log n)\big)$, and
- cannot be solved in time $\exp\big(O(n^{1-1/d-\epsilon}\ell^{1/d})\big)$ for any $\epsilon > 0$, unless the ETH fails.

1 Introduction

On planar graphs, many classic algorithmic problems enjoy a certain "square root phenomenon" and can be solved significantly faster than what is known to be possible on general graphs: for example, Independent Set, 3-Coloring, Hamiltonian Cycle, Dominating Set can be solved in time $2^{O(\sqrt{n})}$ on an n-vertex planar graph, while no $2^{o(n)}$ algorithms exist for general graphs, assuming the Exponential Time Hypothesis (ETH) of Impagliazzo, Paturi, and Zane [2]. The square root in the exponent seems to be best possible for planar graphs: assuming the ETH, the running time for these problems cannot be improved to $2^{o(\sqrt{n})}$.

In some cases, a similar speedup can be obtained for 2-dimensional geometric problems, for example, there are $2^{O(\sqrt{n}\log n)}$ time algorithms for Independent Set on unit disk graphs or for TSP on 2-dimensional point sets [5, 1]. More generally, for d-dimensional geometric problems, running times of the from $2^{O(n^{1-1/d})}$ or $n^{O(k^{1-1/d})}$ appear naturally, and Marx and Sidiropoulos [4] showed that, assuming the ETH, this form of running time is essentially best possible for some problems.

In this paper, we explore whether such a speedup is possible for geometric coloring problems. Let us consider now the problem of coloring the intersection graph of a set of unit disks in the 2-dimensional plane, that is, assigning a color to each disk such that if two disks intersect, then they receive different colors. For a constant number of colors, geometric objects can behave similarly to planar graphs: 3-Coloring can be solved in time $2^{O(\sqrt{n})}$ on the intersection graph of n unit disks in the plane and, assuming the ETH, there is no such algorithm with running time $2^{o(\sqrt{n})}$. However, while every planar graph is 4-colorable, unit disks graphs can contain arbitrary large cliques, and hence the ℓ-colorability is a meaningful question for larger, non-constant, values of ℓ as well. We show that if the number ℓ of colors is part of the input and can be up to $\Theta(n)$, then, surprisingly, no speedup is possible: Coloring the intersection graph of n unit disks with ℓ colors cannot be solved in time $2^{o(n)}$, assuming the ETH. What happens between these two extremes of constant number of colors and $\Theta(n)$ colors? Our main 2-dimensional result exhibits a smooth increase of complexity as the number ℓ of colors increases.

Theorem 1 *For any fixed $0 \leq \alpha \leq 1$, the problem of coloring the intersection graph of n unit disks with $\ell = \Theta(n^\alpha)$ colors*

- *can be solved in time $2^{O(n^{\frac{1+\alpha}{2}}\log n)} = 2^{O(\sqrt{n\ell}\log n)}$, and*
- *cannot be solved in time $2^{o(n^{\frac{1+\alpha}{2}})} = 2^{o(\sqrt{n\ell})}$, unless the ETH fails.*

The proof is not very specific to disks and can be easily adapted to, say, axis-parallel unit squares or other fat objects. However, it seems that the requirement of fatness is essential for this type of complexity behavior as, for example, the coloring of the intersection graphs of line segments (of arbitrary lengths) does not admit

*Department of Mathematics, University of Louisville

†Institute for Computer Science and Control, Hungarian Academy of Sciences (MTA SZTAKI). Supported by the ERC grant PARAMTIGHT: "Parameterized complexity and the search for tight complexity results", no. 280152.

‡Faculty of Mathematics and Information Science, Warsaw University of Technology

any speedup compared to the $2^{O(n)}$ algorithm, even for a constant number of colors.

Theorem 2 *There is no $2^{o(n)}$ time algorithm for 6-*COLORING *the intersection graph of line segments in the plane, unless the ETH fails.*

How does the complexity change if we look at the generalization of the coloring problem into higher dimensions? It is known for some problems that if we generalize the problem from two dimensions to d dimensions, then the square root in the exponent of the running time changes to a $1-1/d$ power, which makes the running time closer and closer to the running time of the brute force as d increases. For the ℓ-coloring problem, the correct exponent seems to be $n^{1-1/d}$ times $\ell^{1/d}$. That is, as d increases, the running time becomes less and less sensitive to the number of colors and approaches $2^{O(n)}$, even for constant number of colors.

Theorem 3 *For any fixed $0 \le \alpha \le 1$ and dimension $d \ge 2$, the problem of coloring the intersection graph of n unit balls in the d-dimensional Euclidean space with $\ell = \Theta(n^\alpha)$ colors*

- *can be solved in time $2^{O\left(n^{\frac{d-1+\alpha}{d}} \log n\right)} = 2^{O\left(n^{1-1/d}\ell^{1/d} \log n\right)}$, and*
- *cannot be solved in time $2^{n^{\frac{d-1+\alpha}{d}-\epsilon}}$ for any $\epsilon > 0$, unless the ETH fails.*

The upper bounds of Theorems 1 and 3 follow fairly easily using standard techniques. Clearly, the problem of coloring unit d-balls with ℓ colors makes sense only if every point of the space is contained in at most ℓ balls: otherwise we would immediately know that there is no ℓ-coloring. On the other hand, if every point is contained in at most ℓ of the n balls, then it is known that there is a balanced separator of size $O(n^{1-1/d}\ell^{1/d})$ [5]. By finding such a separator and trying every possible coloring on the disks of the separator, we can branch into $\ell^{O(n^{1-1/d}\ell^{1/d})}$ smaller instances. This recursive procedure has the running time as claimed.

2 Auxiliary problems

We start with introducing two auxiliary problems, which will serve as middle steps in the hardness reduction. For a fixed dimension d and $i \in [d]$, we denote by e_i the d-dimensional vector, whose i-th coordinate is equal to 1 and all remaining coordinates are equal to 0. For two positive integers g, d, we denote by $R[g, d]$ the *d-dimensional grid*, i.e., a graph whose vertices are all vectors from $[g]^d$, and two vertices are adjacent if they differ on exactly one coordinate, and exactly by one (on that coordinate). In other words, a and a' are adjacent if $a = a' \pm e_i$ for some $i \in [d]$. We will often refer to vertices of a grid as *cells*.

Problem: d-GRID 3-SAT
Input: A d-dimensional grid $G = R[g, d]$, $k \in \mathbb{N}_+$, a function $\zeta : v \in V(G) \mapsto \{v_1, v_2, \dots, v_k\}$ mapping each cell v to k fresh boolean variables, and a set $\mathcal{C}$ of constraints of two kinds:

clause constraints: for a cell v, a set $\mathcal{C}(v)$ of pairwise variable-disjoint disjunctions of at most 3 literals on $\zeta(v)$;
equality constraints: for adjacent cells v and w, a set $\mathcal{C}(v, w)$ of pairwise variable-disjoint constraints of the form $v_i = w_j$ (with $i, j \in [k]$).

Question: Is there an assignment of the variables such that all constraints are satisfied?

The size of the instance $I = (G, k, \zeta, \mathcal{C})$ of d-GRID 3-SAT is the total number of variables, i.e., $g^d k$.

Problem: PARTIAL d-GRID COLORING
Input: An induced subgraph G of $R[g, d]$, $\ell \in \mathbb{N}_+$, and a function $\rho : v \in V(G) \mapsto \{p_1^v, p_2^v, \dots, p_\ell^v\} \in ([\ell]^d)^\ell$ mapping each cell v to a set of ℓ points in $[\ell]^d$.
Question: Is there an ℓ-coloring of all the points such that:

- two points in the same cell get different colors;
- if v and w are adjacent in G, say, $w = v + e_i$ (for some $i \in [d]$), and $p \in \rho(v)$ and $q \in \rho(w)$ receive the same color, then $p[i] \leqslant q[i]$ where $a[i] := a \cdot e_i$ is the i-th coordinate of a?

Here the size of the instance is the total number of points, i.e., $|V(F)|\ell \le g^d \ell$.

3 2-dimensional lower bounds

First, by a reduction from 3-SAT, we show that 2-GRID 3-SAT with total size n and k variables per cell cannot be solved in time $2^{o(\sqrt{nk})}$, unless the ETH fails. The main result of this section is the following theorem.

Theorem 4 *For any $0 \leqslant \alpha \leqslant 1$, there is no $2^{o(\sqrt{n\ell})}$ algorithm solving* PARTIAL 2-GRID COLORING *on a total of n points and $\ell = \Theta(n^\alpha)$ points in each cell (that is n/ℓ cells), unless the ETH fails.*

Proof. We present a reduction from 2-GRID 3-SAT to PARTIAL 2-GRID COLORING. Let $I = (G, k, \zeta, \mathcal{C})$ be an instance of 2-GRID 3-SAT, where $G = R[g, 2]$ and each cell contains k variables. We construct an equivalent instance $J = (F, \ell, \rho)$ of PARTIAL 2-GRID COLORING with $|V(F)| = \Theta(|V(G)|) = \Theta(g^2)$ and $\ell := 4k$ points per cell, where F is an induced subgraph of $R[g', 2]$ with $g' = \Theta(g)$. Let us present the key ideas of the construction.

Standard cells. A *standard cell* is a cell where the points $p_1, \dots, p_\ell$ are on the main diagonal, that is $p_i = (i, i)$ for every $i \in [\ell]$ (see cells in Fig. 1).

Reference coloring. Later in the construction we will choose one standard cell $\bar{R}$, whose coloring will be referred to as the *reference coloring*. Now, by the color $i \in [\ell]$, we mean the color of the point p_i in $\bar{R}$.

Variable-assignment cells. For each cell $v = (i, j) \in V(G)$, we introduce in F a standard cell $A(v)$), called the *variable-assignment cell*. The cell $A(v)$ is responsible for encoding the truth assignment of variables in $\zeta(v)$. If $i + j$ is even, then the cell $A(v)$ is also called *even*. Otherwise $A(v)$ is *odd*.

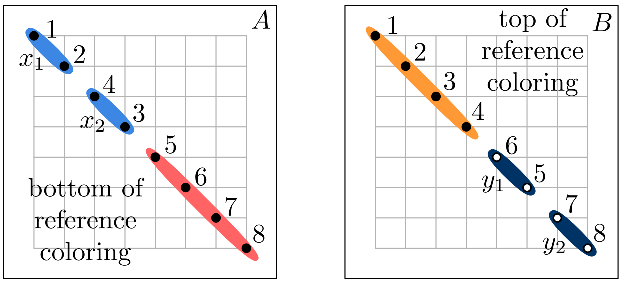

Figure 1: Even cells (left) and odd (right) variable-assignment cells.

In our construction, we make sure that each variable-assignment cell receives one of the *standard colorings*. If $A(v)$ is even, the coloring φ of $A(v)$ is standard if $\{\varphi(p_{2i-1}), \varphi(p_{2i})\} = \{2i-1, 2i\}$ for $i \in [k]$ and $\varphi(p_i) = i$ for $i \in [4k] \setminus [2k]$. If the cell $A(v)$ is odd, its standard colorings φ are the ones with $\varphi(p_i) = i$ for $i \in [2k]$ and $\{\varphi(p_{2i-1}), \varphi(p_{2i})\} = \{2i-1, 2i\}$ for $i \in [2k] \setminus [k]$. The choice of the particular standard coloring for the points in $A(v)$ defines the actual assignment of variables in $\zeta(v)$. If $A(v)$ is even, then for each $i \in [k]$, we interpret the coloring in the following way:

$p_{2i-1} \mapsto 2i-1$, $p_{2i} \mapsto 2i$ as setting the variable v_i to true;

$p_{2i-1} \mapsto 2i$, $p_{2i} \mapsto 2i-1$ as setting the variable v_i to false.

If $A(v)$ is odd, for each $i \in [k]$, we interpret it in that way:

$p_{2k+2i-1} \mapsto 2i-1$, $p_{2k+2i} \mapsto 2i$ as setting the variable v_i to true;

$p_{2k+2i-1} \mapsto 2i$, $p_{2k+2i} \mapsto 2i-1$ as setting the variable v_i to false.

Local reference cells. For each inner face of G (see Fig. 2), we introduce a new standard cell, called a *local reference cell*. Moreover, we set the reference $\bar{R}$ to be uppermost-leftmost local reference cell. In the construction, we will ensure that the coloring of each local reference cell is exactly the same, i.e., is exactly the reference coloring.

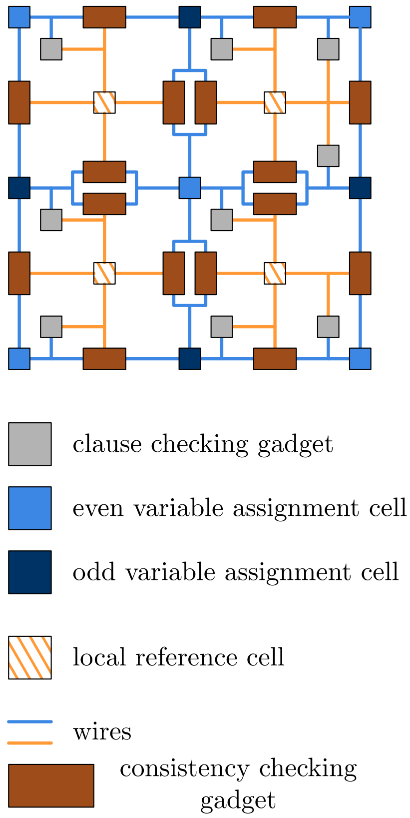

Figure 2: High-level illustration of J.

Overview of the construction. Figure 2 presents the arrangement of the cells in F. For each variable-assignment cell $A(v)$, we introduce a *clause-checking gadget*, which is responsible for ensuring that all clauses in $\mathcal{C}(v)$ are satisfied. This gadget requires an access to the reference coloring, which can attain from the local reference cells (we can choose any of the local reference cells close to $A(v)$). For each edge vw of G, we introduce a *consistency gadget*. In fact, for inner edges of G (i.e., the ones not incident with the outer face) we introduce two consistency gadgets, one for each face incident with vw. This gadget is responsible for ensuring the consistency on three different levels:

- to force all equality constraints $\mathcal{C}(v, w)$ to be satisfied,

- to ensure that each of $A(v)$ and $A(w)$ receives one of the standard colorings,
- to ensure that the local reference cell contains exactly the reference coloring.

This gadget also requires access to the reference coloring, so we join it with the appropriate local reference cell (see Fig. 2).

Now, we observe that the total number of points in F is $n = O(g^2\ell) = O(n')$, where $n' = g^2k$ is the total size of I. Thus, the existence of an algorithm solving J in time $2^{o(\sqrt{n\ell})}$ could be used to solve I in time $2^{o(\sqrt{n'k})}$, which, in turn, contradicts the ETH. $\square$

Now, to prove the lower bound in Theorem 1, we need to show a reduction from Partial 2-grid Coloring to the problem of coloring unit disk graphs. This reduction is fairly standard and uses a well-known approach [3, Theorems 1 and 3]. presented on Figure 3.

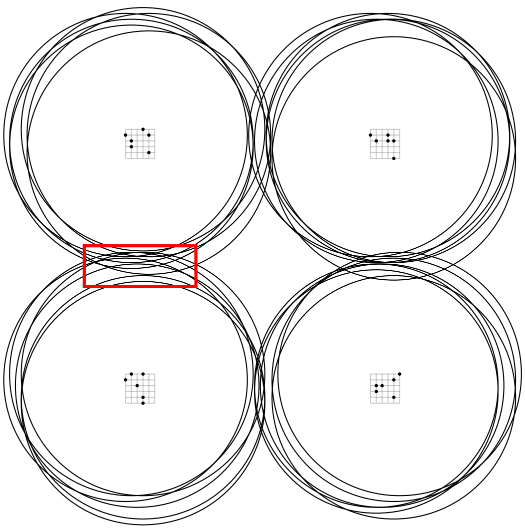

Figure 3: Reduction from Partial 2-grid Coloring to coloring unit disks.

4 Coloring unit d-dimensional balls

The d-dimensional lower bound of Theorem 3 goes along the same lines, but we first prove a lower bound for d-grid 3-Sat. Based on earlier results by Marx and Sidiropoulos [4], we prove an almost tight lower bound for this d-dimensional 3-SAT by embedding a 3-SAT instance with roughly $g^{d-1}k$ variables and clauses into the d-dimensional $g \times \cdots \times g$-grid $R[g, d]$. Then the reduction from this problem to coloring unit balls in d-dimensional space is very similar to the 2-dimensional case.

References

[1] J. Alber and J. Fiala. Geometric separation and exact solutions for the parameterized independent set problem on disk graphs. *J. Algorithms*, 52(2):134–151, 2004.

[2] R. Impagliazzo, R. Paturi, and F. Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001.

[3] D. Marx. Efficient approximation schemes for geometric problems? In *ESA 2005 Proc.*, pages 448–459, 2005.

[4] D. Marx and A. Sidiropoulos. The limited blessing of low dimensionality: When 1-1/d is the best possible exponent for d-dimensional geometric problems. SOCG 2014 Proc., pages 67:67–67:76, New York, NY, USA, 2014. ACM.

[5] W. D. Smith and N. C. Wormald. Geometric separator theorems and applications. FOCS 1998 Proc., pages 232–, Washington, DC, USA, 1998. IEEE Computer Society.