Recognition of Unit Segment and Polyline Graphs is $\exists\mathbb{R}$-Complete*

Michael Hoffmann[1], Tillmann Miltzow[2], Simon Weber[1], and Lasse Wulf[3]

1 **Department of Computer Science, ETH Zürich, Switzerland**
2 **Department of Information and Computing Sciences, Utrecht University, The Netherlands**
3 **Institute of Discrete Mathematics, Graz Institute of Technology, Austria**

Abstract

Given a set of objects O in the plane, the corresponding intersection graph is defined as follows. A vertex is created for each object and an edge joins two vertices whenever the corresponding objects intersect. We study here the case of unit segments and polylines with exactly k bends. In the recognition problem, we are given a graph and want to decide whether the graph can be represented as the intersection graph of certain geometric objects. In previous work it was shown that various recognition problems are $\exists\mathbb{R}$-complete, leaving unit segments and polylines as few remaining natural cases. We show that recognition for both families of objects is $\exists\mathbb{R}$-complete.

Related Version *Full Version*: `arXiv:2401.02172`

1 Introduction

Many real-life problems can be mathematically described in the language of graphs. For instance, Cellnex Telecom owns more than 2000 cell towers in Switzerland. We want to assign each tower a frequency such that no two towers that overlap in coverage use the same frequency. This becomes a graph coloring problem. Every cell tower becomes a vertex, overlap indicates an edge and a frequency assignment corresponds to a proper coloring of the vertices, see Figure 1.

In many contexts, we have additional structure on the graph that may or may not help us to solve the underlying algorithmic problem. For instance, it might be that the graph arises as the intersection graph of unit disks in the plane (each unit disk gives a vertex, and two vertices are adjacent if their corresponding disks overlap). In that case, the coloring problem can be solved more efficiently [10], and there are better approximation algorithms for the clique problem [11]. This motivates a systematic study of geometric intersection graphs.

It is known for a host of geometric shapes that it is $\exists\mathbb{R}$-complete to recognize their intersection graphs [28, 14, 25, 27]. The class $\exists\mathbb{R}$ consists of all of those problems that are polynomial-time equivalent to deciding whether a polynomial $p \in \mathbb{Z}[X_1, \ldots, X_n]$ has a root. We will introduce $\exists\mathbb{R}$ in more detail below. $\exists\mathbb{R}$-completeness is known for the recognition problems of intersection graphs for segments, disks, unit disks, rays, grounded segments, downward rays, and a few other examples.

* This research started at the *19th Gremo Workshop on Open Problems* in Binn VS, Switzerland, in June 2022. We thank the organizers for the invitation and for providing a very pleasant and inspiring working atmosphere. Tillmann Miltzow is generously supported by the Netherlands Organisation for Scientific Research (NWO) under project no. VI.Vidi.213.150. Lasse Wulf is generously supported by the Austrian Science Fund (FWF): W1230.

40th European Workshop on Computational Geometry, Ioannina, Greece, March 13–15, 2024.
This is an extended abstract of a presentation given at EuroCG'24. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

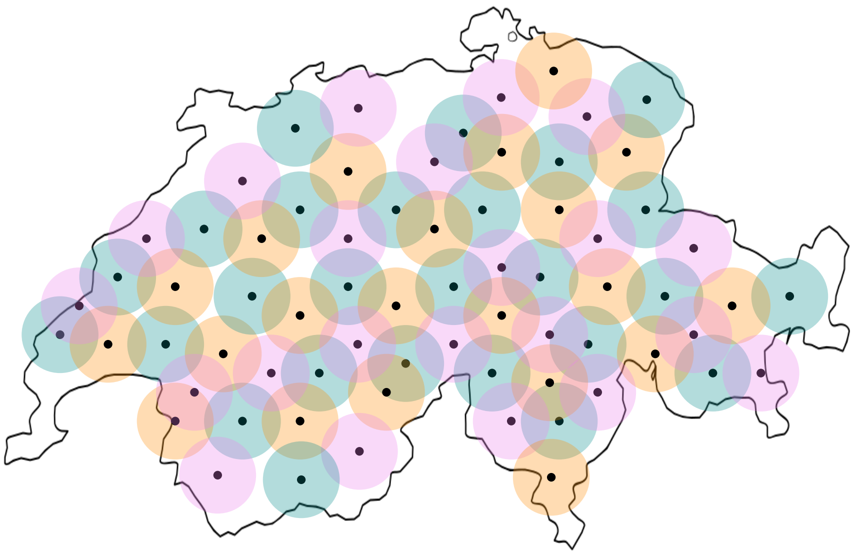

Figure 1 A fictional illustration of mobile coverage of Switzerland using cell towers.

In this work, we focus on two geometric objects; unit segments and polylines with exactly k bends. Although we consider both types of geometric objects natural and well studied, to the best of our knowledge the complexity of their recognition problem was left open.

1.1 Definition and Results

Given a finite set of geometric objects O, we denote by $G(O) = (V, E)$, the corresponding *intersection graph*. The set of vertices is the set of objects ($V = O$) and two objects are adjacent ($uv \in E$) if they intersect ($u \cap v \neq \emptyset$). We are interested in intersection graphs that come from different families of geometric objects.

Examples for a family of geometric objects are segments, disks, unit disks, unit segments, rays, and convex sets, to name a few of the most common ones. In general, given a geometric body $O \subset \mathbb{R}^2$ we denote by O^{+} the family of all translates of O. Similarly, we denote by $O^{\oplus}$ the family of all translates and rotations of O. For example, the family of all unit segments can be denoted as $u^{\oplus}$, where u is a unit segment.

Classes of geometric objects $\mathcal{O}$ naturally give rise to classes of graphs $C(\mathcal{O})$: Given a family of geometric objects $\mathcal{O}$, we denote by $C(\mathcal{O})$ the class of graphs that can be formed by taking the intersection graph of a finite subset from $\mathcal{O}$.

If we are given a graph, we can ask if this graph belongs to a geometric graph class. Formally, let C be a fixed graph class, then the recognition problem for C is defined as follows. As input, we receive a graph G and we have to decide whether $G \in C$. We denote the corresponding algorithmic problem by Recognition(C). For example the problem of recognizing unit segment graphs can be denoted by Recognition($C(u^{\oplus})$). We will use the term Unit Recognition for this problem. Furthermore, we define PolyLine Recognition as the recognition problem of intersection graphs of polylines with k bends.

We can also say that Recognition($C(\mathcal{O})$) asks about the existence of a *representation* of a given graph. A representation or *realization* of a graph G using a family of objects $\mathcal{O}$ is a function $r : V \mapsto \mathcal{O}$ such that $r(v) \cap r(w) \neq \emptyset \iff vw \in E$. For simplicity, for a set $V' \subseteq V$, we define $r(V') = \bigcup_{v \in V'} r(v)$.

Results. We show ∃ℝ-completeness of the recognition problems of two very natural geometric graph classes.

▶ **Theorem 1.1.** *Unit Recognition is ∃ℝ-complete.*

▶ **Theorem 1.2.** *PolyLine Recognition is ∃ℝ-complete, for any fixed $k \geq 1$.*

1.2 Discussion

To supply the appropriate context for our results, we give a comprehensive overview over important geometric graph classes and the current knowledge about the complexity of their recognition problems in Figure 2.

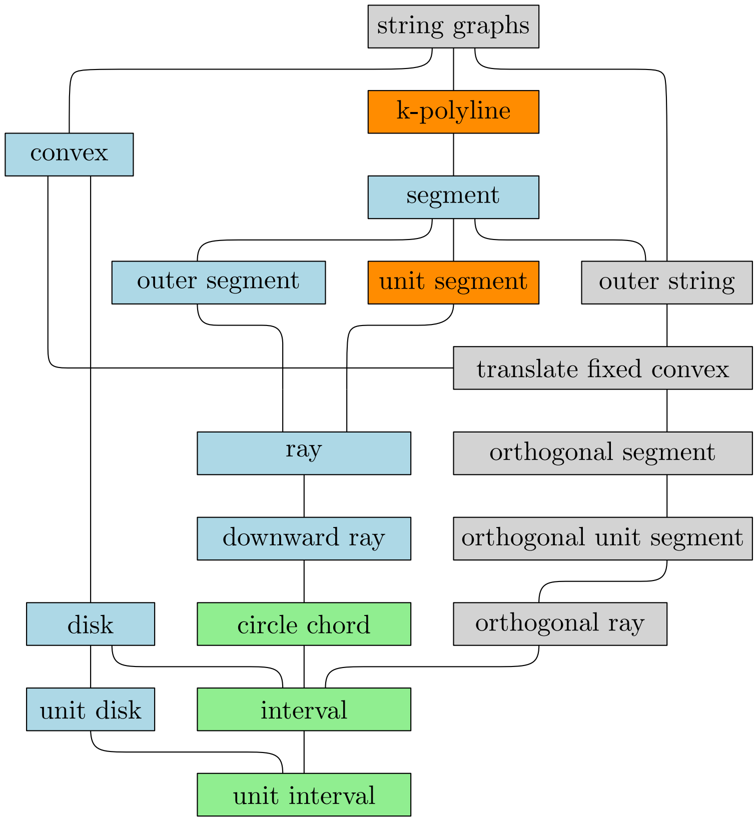

Figure 2 Each box represents a different geometric intersection graph class. Those marked in green can be recognized in polynomial time. Those in blue are known to be ∃ℝ-complete. The ones in gray are NP-complete, and the orange ones are the new results presented in this paper. Relevant references: [12, 14, 21, 22, 23, 24, 25, 27, 28, 30, 32, 33, 36, 43]

Refining the Hierarchy. We see our main contribution in refining the hierarchy of geometric graph classes for which recognition complexity is known. Both unit segments as well as polylines with k bends are natural objects that are well studied in the literature. However, the recognition of the corresponding graph classes was not studied previously. Polylines with an unbounded number of bends are equivalent to strings[1], while polylines

[1] It is possible to show that polylines with an unbounded number of bends are as versatile as strings with respect to the types of graphs that they can represent, since the number of intersections of any two strings can always be bounded from above [36, 37].

with 0 bends are just segments. Polylines with k bends thus naturally slot in between strings and segments, and their corresponding graph class is thus also an intermediate class between the class of segment intersection graphs and string graphs, as can be seen in Figure 2. By showing that recognition for polylines with k bends is ∃ℝ-complete for all constant k, we see that the switch from ∃ℝ-completeness (segment intersection graphs) to NP-membership (string graphs) really only happens once k is infinite. Similarly, unit segment intersection graphs slot in between segment and ray intersection graphs. Intuitively, recognition of a class intermediate to two classes that are ∃ℝ-hard to recognize should also be ∃ℝ-hard, and our Theorem 1.1 confirms this intuition in this case.

Large Coordinates. One of the consequences of ∃ℝ-completeness is that there are no short representations of solutions known. Some representable graphs may only be representable by objects with irrational coordinates, or by rational coordinates with nominator and denominator of size at least $2^{2^{n^c}}$, for some fixed $c > 0$. In other words, the numbers to describe the position might need to be doubly exponentially large [27] for some graphs. For "flexible" objects like polylines, rational solutions can always be obtained by slightly perturbing the representation. For more "sturdy" objects like unit segments this may not be possible, however it is known that for example unit disks admit rational solutions as well [28].

Unraveling the Broader Story. Given the picture of Figure 2, we wish to get a better understanding of when geometric graph recognition problems are ∃ℝ-complete and when they are contained in NP. Figure 2 indicates that ∃ℝ-hardness comes from objects that are complicated enough to avoid a complete combinatorial characterization. Such characterizations are known for example for unit interval graphs, interval graphs and circle chord graphs. On the other hand, if the geometric objects are too flexible, the recognition problem is in NP. The prime example is string graphs [36]. We want to summarize this as: recognition problems are ∃ℝ-complete if the underlying family of geometric objects is at a sweet spot of neither being too simplistic nor too flexible.

Studying the figure further we observe two different types of ∃ℝ-complete families. The first type of family encapsulates all rotations $O^{\circlearrowleft}$ of a given object O (i.e., segments, rays, unit segments etc.). The second type of family contains translates and possibly homothets of geometric objects that have some curvature themselves (i.e., disks and unit disks). However in case we fix a specific object without curvature, i.e., a polygon, and consider all translations of it then the recognition problem also lies in NP [30]. Therefore, broadly speaking, curvature or rotation seem to be properties needed for ∃ℝ-completeness and the lack of it seems to imply NP-membership. We wish to capture parts of this intuition in the following conjectures:

▶ Conjecture 1. Let O be a convex body in the plane with at least two distinct points. Then Recognition($O^{\circlearrowleft}$) is ∃ℝ-complete.

▶ Conjecture 2. Let O be a convex body in the plane. Then Recognition(O^{+}) is ∃ℝ-complete if and only if O has curvature.

1.3 Existential Theory of the Reals

The class of the existential theory of the reals ∃ℝ (pronounced as 'ER') is a complexity class defined through its canonical problem ETR, which also stands for Existential Theory of the Reals. In this problem we are given a sentence of the form $\exists x_1, \ldots, x_n \in \mathbb{R} : \Phi(x_1, \ldots, x_n)$, where Φ is a well-formed quantifier-free formula consisting of the symbols $\{0, 1, x_1, \ldots, x_n, +, \cdot, \geq, >, \wedge, \vee, \neg\}$, and the goal is to check whether this sentence is true.

The class ∃ℝ is the family of all problems that admit a polynomial-time many-one reduction to ETR. It is known that NP $\subseteq$ ∃ℝ $\subseteq$ PSPACE [13]. The reason that ∃ℝ is an important complexity class is that a number of common problems, mainly in computational geometry, have been shown to be complete for this class. Schaefer established the current name and pointed out first that several known NP-hardness reductions actually imply $\exists\mathbb{R}$-completeness [33]. Early examples are related to recognition of geometric structures: points in the plane [29, 42], geometric linkages [34, 1], segment graphs [25, 27], unit disk graphs [28, 21], ray intersection graphs [14], and point visibility graphs [14]. In general, the complexity class is more established in the graph drawing community [26, 16, 35, 18]. Yet, it is also relevant for studying polytopes [31, 17], Nash-Equilibria [6, 38, 20, 8, 9], and matrix factorization problems [15, 40, 41, 39]. Other $\exists\mathbb{R}$-complete problems are the Art Gallery Problem [3, 44], covering polygons with convex polygons [2], geometric packing [5] and training neural networks [4, 7].

2 Proof Techniques

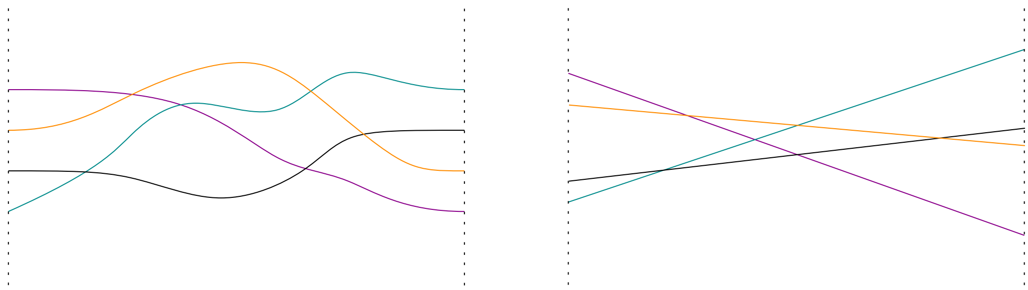

Figure 3 The pseudoline arrangement on the left is combinatorially equivalent to the (truncated) line arrangement on the right; hence, it is stretchable.

The techniques used in this paper are similar to previous work. Due to space constraints we only give some rough proof sketches, all the details can be found in the full version of the paper. $\exists\mathbb{R}$-membership can be established straightforwardly by constructing concrete formulae or invoking a characterization of $\exists\mathbb{R}$ using *real* verification algorithms, similar to the characterization of NP [19]. For $\exists\mathbb{R}$-hardness, we are in essence reducing from the SimpleStretchability problem. In this problem, we are given a simple *pseudoline arrangement* as an input, and the question is whether this arrangement is *stretchable*. A pseudoline arrangement $\mathcal{A}$ is a set of n curves that are x-monotone. Furthermore, any two curves intersect exactly once and no three curves meet in a single point. We assume that there exist two vertical lines on which each curve starts and ends. The problem is to determine whether there exists a combinatorially equivalent (truncated) line arrangement. See Figure 3 for an example.

Given the initial pseudoline arrangement $\mathcal{A}$, we construct a graph that is representable by unit segments iff $\mathcal{A}$ is stretchable. This graph is created by enhancing $\mathcal{A}$ with more curves (see Figure 4) and taking their intersection graph. Figures 5 to 7 give some intuition on the proof that if $\mathcal{A}$ is stretchable, this graph is representable by unit segments: The line arrangement certifying stretchability is first squeezed into a canonical form, then all features can be represented easily. On the other hand, if this graph is representable, the unit segments representing the vertices corresponding to $\mathcal{A}$ witness stretchability of $\mathcal{A}$. To prove this, we show that cycles can be used to enforce a certain order of intersections of objects with the cycle. For this, we can use the same proof for unit segments and polylines.

Knowing that the connectors (green in Figure 4) must intersect the cycle in the correct

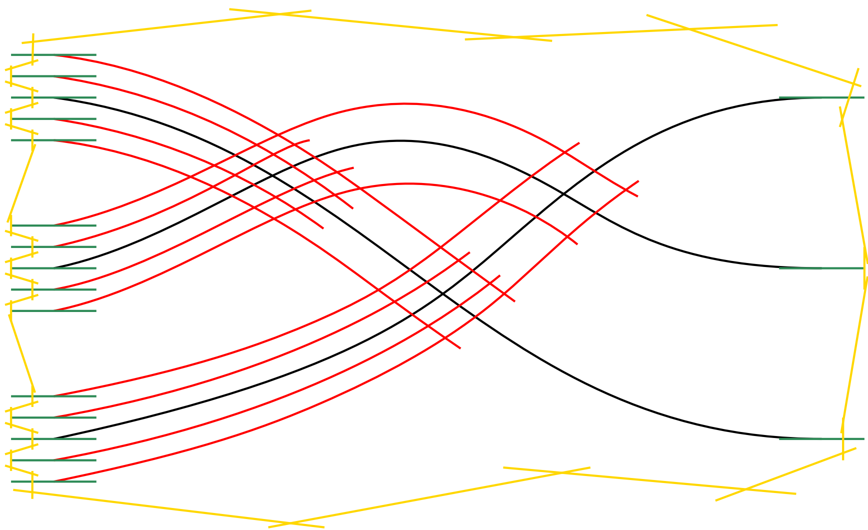

Figure 4 $\mathcal{A}$ (black), enhanced with probes (red), connectors (green) and a cycle (yellow).

order in any representation, the intersection pattern of the pseudolines and the probes (red) guarantees that unit segments representing the pseudolines must have the same combinatorial structure as $\mathcal{A}$, finishing the proof. The ideas of order-enforcing cycles and probes have already been used in different contexts [14].

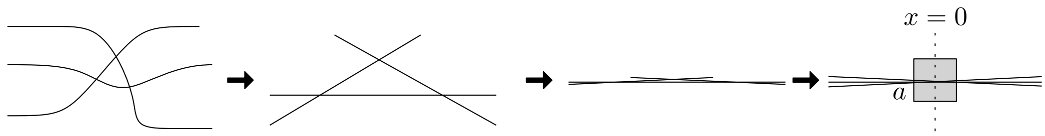

Figure 5 Stretching and then squeezing a pseudoline arrangement.

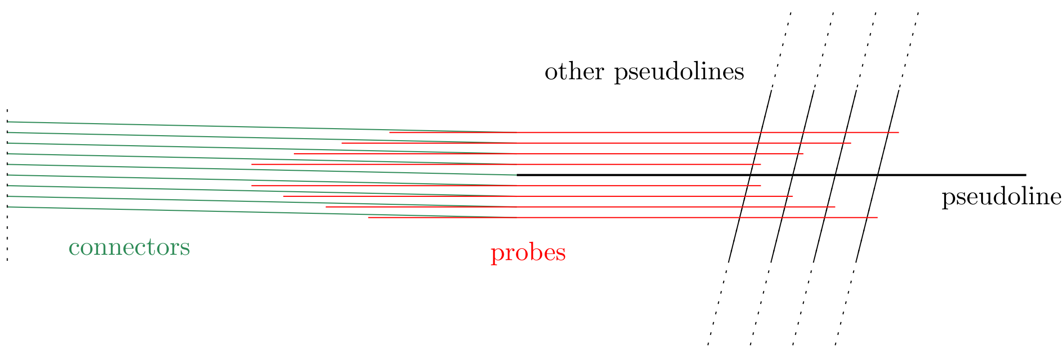

Figure 6 Representing connectors and probes by unit segments.

The ∃ℝ-hardness proof for PolyLine Recognition follows this previous proof for unit segments closely, and only adds some additional order-enforcing cycles. The enhanced pseudoline arrangement for polylines is shown in Figure 8. For each pseudoline we create a twin. Using the $2k$ additional order-enforcing cycles we enforce that in any realization of the graph by polylines, each polyline representing a pseudoline must intersect its twin within that cycle. We can then show that if two polylines intersect $2k$ times (with both polylines

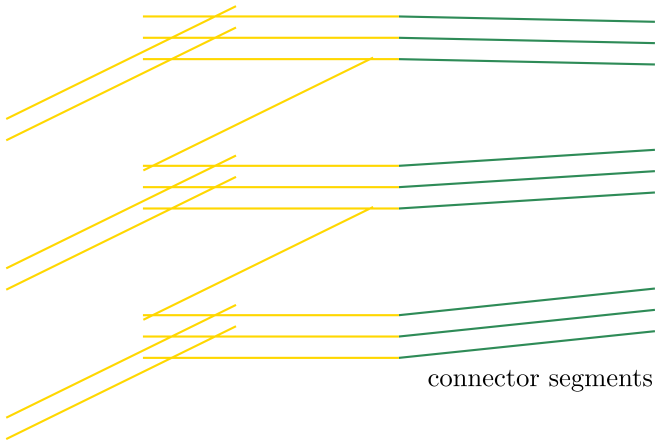

Figure 7 Attaching the connectors to the cycle using unit segments in a sawtooth pattern.

visiting these intersection points in the same order), they must in total use at least $2k-1$ bends. This ensures that at least one of the k-polylines must spend all of its k bends to realize these intersections. Thus, this polyline is actually a straight line in the region labelled "canvas" in Figure 8. Using the same arguments as for unit segments we can then see that the arrangement formed by these straight lines is combinatorially equivalent to $\mathcal{A}$, and thus $\mathcal{A}$ is stretchable.

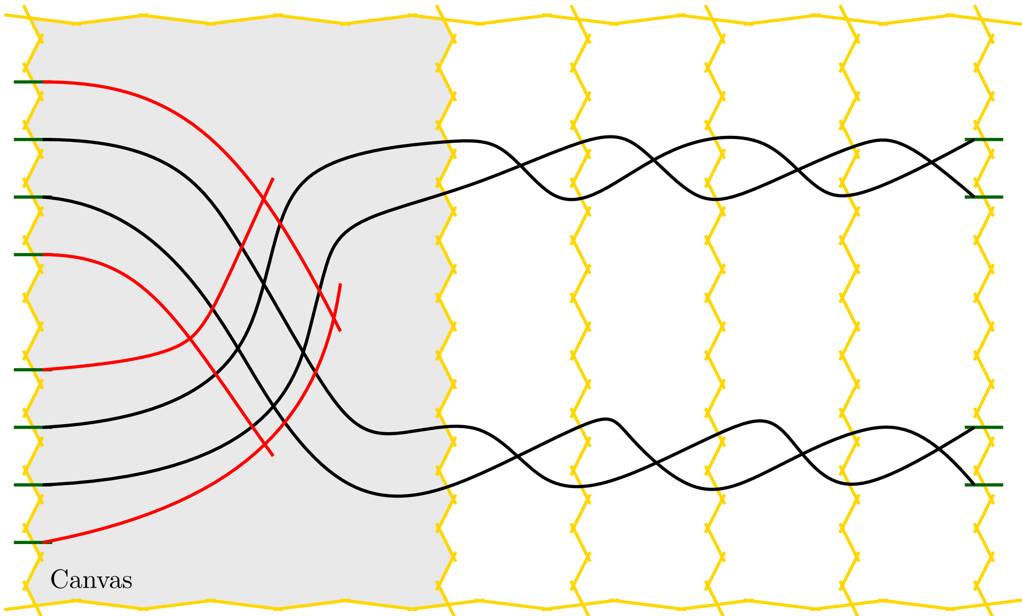

Figure 8 $\mathcal{A}$ *twinned* and enhanced with probes, connectors and $2k+1$ cycles (yellow). Weaving the twinned pseudolines ensures that at least one of the two twins contains no bends in the canvas.

References

1 Zachary Abel, Erik Demaine, Martin Demaine, Sarah Eisenstat, Jayson Lynch, and Tao Schardl. Who needs crossings? Hardness of plane graph rigidity. In *32nd International Symposium on Computational Geometry (SoCG 2016)*, pages 3:1–3:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.SoCG.2016.3.

2 Mikkel Abrahamsen. Covering Polygons is Even Harder. In Nisheeth K. Vishnoi, editor, *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 375–386, 2022. doi:10.1109/FOCS52979.2021.00045.

3 Mikkel Abrahamsen, Anna Adamaszek, and Tillmann Miltzow. The art gallery problem is ∃ℝ-complete. *Journal of the ACM*, 69(1):4:1–4:70, 2022. doi:10.1145/3486220.

4 Mikkel Abrahamsen, Linda Kleist, and Tillmann Miltzow. Training neural networks is ∃ℝ-complete. In *NeurIPS 2021*, pages 18293–18306, 2021. URL: https://proceedings.neurips.cc/paper/2021/hash/9813b270ed0288e7c0388f0fd4ec68f5-Abstract.html.

5 Mikkel Abrahamsen, Tillmann Miltzow, and Nadja Seiferth. Framework for ∃ℝ-completeness of two-dimensional packing problems. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1014–1021, 2020. doi:10.1109/FOCS46700.2020.00098.

6 Marie Louisa Tølbøll Berthelsen and Kristoffer Arnsfelt Hansen. On the computational complexity of decision problems about multi-player Nash equilibria. *Theory of Computing Systems*, 66(3):519–545, 2022. doi:10.1007/s00224-022-10080-1.

7 Daniel Bertschinger, Christoph Hertrich, Paul Jungeblut, Tillmann Miltzow, and Simon Weber. Training fully connected neural networks is ∃ℝ-complete. *CoRR*, abs/2204.01368, 2022. To appear at NeurIPS 2024. arXiv:2204.01368.

8 Vittorio Bilò and Marios Mavronicolas. A catalog of ∃ℝ-complete decision problems about Nash equilibria in multi-player games. In *33rd Symposium on Theoretical Aspects of Computer Science (STACS 2016)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.STACS.2016.17.

9 Vittorio Bilò and Marios Mavronicolas. ∃ℝ-complete decision problems about symmetric Nash equilibria in symmetric multi-player games. In *34th Symposium on Theoretical Aspects of Computer Science (STACS 2017)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.STACS.2017.13.

10 Csaba Biró, Édouard Bonnet, Dániel Marx, Tillmann Miltzow, and Pawel Rzazewski. Fine-grained complexity of coloring unit disks and balls. *Journal of Computational Geometry*, 9(2):47–80, 2018. doi:10.20382/jocg.v9i2a4.

11 Marthe Bonamy, Édouard Bonnet, Nicolas Bousquet, Pierre Charbit, Panos Giannopoulos, Eun Jung Kim, Pawel Rzazewski, Florian Sikora, and Stéphan Thomassé. EPTAS and subexponential algorithm for maximum clique on disk and unit ball graphs. *Journal of the ACM*, 68(2):9:1–9:38, 2021. doi:10.1145/3433160.

12 Kellogg S. Booth and George S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using *PQ*-tree algorithms. *Journal of Computer and System Sciences*, 13(3):335–379, 1976. doi:10.1016/S0022-0000(76)80045-1.

13 John Canny. Some algebraic and geometric computations in PSPACE. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 460–467, New York, NY, USA, 1988. Association for Computing Machinery. doi:10.1145/62212.62257.

14 Jean Cardinal, Stefan Felsner, Tillmann Miltzow, Casey Tompkins, and Birgit Vogtenhuber. Intersection Graphs of Rays and Grounded Segments. *Journal of Graph Algorithms and Applications*, 22(2):273–294, 2018. doi:10.7155/jgaa.00470.

15 Dmitry Chistikov, Stefan Kiefer, Ines Marusic, Mahsa Shirmohammadi, and James Worrell. On Restricted Nonnegative Matrix Factorization. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium*

on Automata, Languages, and Programming (ICALP 2016), volume 55 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 103:1–103:14, Dagstuhl, Germany, 2016. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ICALP.2016.103.

16 Michael G. Dobbins, Linda Kleist, Tillmann Miltzow, and Paweł Rzążewski. ∀∃ℝ-Completeness and Area-Universality. In Andreas Brandstädt, Ekkehard Köhler, and Klaus Meer, editors, *Graph-Theoretic Concepts in Computer Science (WG 2018)*, volume 11159 of *Lecture Notes in Computer Science*, pages 164–175. Springer, 2018. doi:10.1007/978-3-030-00256-5_14.

17 Michael Gene Dobbins, Andreas Holmsen, and Tillmann Miltzow. A universality theorem for nested polytopes. *arXiv*, 1908.02213, 2019.

18 Jeff Erickson. Optimal curve straightening is ∃ℝ-complete. *arXiv:1908.09400*, 2019.

19 Jeff Erickson, Ivor van der Hoog, and Tillmann Miltzow. Smoothing the gap between NP and ER. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1022–1033, 2020. doi:10.1109/FOCS46700.2020.00099.

20 Jugal Garg, Ruta Mehta, Vijay V. Vazirani, and Sadra Yazdanbod. ∃ℝ-completeness for decision versions of multi-player (symmetric) Nash equilibria. *ACM Transactions on Economics and Computation*, 6(1), 2018. doi:10.1145/3175494.

21 Ross Kang and Tobias Müller. Sphere and Dot Product Representations of Graphs. *Discrete & Computational Geometry*, 47(3):548–569, 2012. doi:10.1007/s00454-012-9394-8.

22 Jan Kratochvíl. String graphs. I. The number of critical nonstring graphs is infinite. *Journal of Combinatorial Theory. Series B*, 52(1):53–66, 1991. doi:10.1016/0095-8956(91)90090-7.

23 Jan Kratochvíl. String graphs. II. Recognizing string graphs is NP-hard. *Journal of Combinatorial Theory. Series B*, 52(1):67–78, 1991. doi:10.1016/0095-8956(91)90091-W.

24 Jan Kratochvíl and Jiří Matoušek. NP-hardness results for intersection graphs. *Commentationes Mathematicae Universitatis Carolinae*, 30(4):761–773, 1989.

25 Jan Kratochvíl and Jiří Matoušek. Intersection graphs of segments. *Journal of Combinatorial Theory. Series B*, 62(2):289–315, 1994. doi:10.1006/jctb.1994.1071.

26 Anna Lubiw, Tillmann Miltzow, and Debajyoti Mondal. The complexity of drawing a graph in a polygonal region. *Journal of Graph Algorithms and Applications*, 26(4):421–446, 2022. doi:10.7155/jgaa.00602.

27 Jiří Matoušek. Intersection graphs of segments and ∃ℝ. *ArXiv 1406.2636*, 2014.

28 Colin McDiarmid and Tobias Müller. Integer realizations of disk and segment graphs. *Journal of Combinatorial Theory. Series B*, 103(1):114–143, 2013. doi:10.1016/j.jctb.2012.09.004.

29 Nicolai Mnëv. The universality theorems on the classification problem of configuration varieties and convex polytopes varieties. In *Topology and geometry – Rohlin seminar*, pages 527–543, 1988.

30 Tobias Müller, Erik Jan van Leeuwen, and Jan van Leeuwen. Integer representations of convex polygon intersection graphs. *SIAM Journal on Discrete Mathematics*, 27(1):205–231, 2013. doi:10.1137/110825224.

31 Jürgen Richter-Gebert and Günter M. Ziegler. Realization spaces of 4-polytopes are universal. *Bulletin of the American Mathematical Society*, 32(4):403–412, 1995.

32 Alexandre Rok and Bartosz Walczak. Outerstring graphs are χ-bounded. *SIAM Journal on Discrete Mathematics*, 33(4):2181–2199, 2019. doi:10.1137/17M1157374.

33 Marcus Schaefer. Complexity of some geometric and topological problems. In David Eppstein and Emden R. Gansner, editors, *Graph Drawing*, pages 334–344, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. doi:10.1007/978-3-642-11805-0_32.

34 Marcus Schaefer. *Realizability of Graphs and Linkages*, pages 461–482. Thirty Essays on Geometric Graph Theory. Springer, 2013. doi:10.1007/978-1-4614-0110-0_24.

35 Marcus Schaefer. Complexity of geometric k-planarity for fixed k. *Journal of Graph Algorithms and Applications*, 25(1):29–41, 2021. doi:10.7155/jgaa.00548.

36 Marcus Schaefer, Eric Sedgwick, and Daniel Štefankovič. Recognizing string graphs in NP. *Journal of Computer and System Sciences*, 67(2):365–380, 2003. Special Issue on STOC 2002. doi:10.1016/S0022-0000(03)00045-X.

37 Marcus Schaefer and Daniel Štefankovič. Decidability of string graphs. *Journal of Computer and System Sciences*, 68(2):319–334, 2004. Special Issue on STOC 2001. doi:10.1016/j.jcss.2003.07.002.

38 Marcus Schaefer and Daniel Štefankovič. Fixed points, Nash equilibria, and the existential theory of the reals. *Theory of Computing Systems*, 60(2):172–193, Feb 2017. doi:10.1007/s00224-015-9662-0.

39 Marcus Schaefer and Daniel Štefankovič. The complexity of tensor rank. *Theory of Computing Systems*, 62(5):1161–1174, 2018. doi:10.1007/s00224-017-9800-y.

40 Yaroslav Shitov. A universality theorem for nonnegative matrix factorizations. *ArXiv 1606.09068*, 2016.

41 Yaroslav Shitov. The complexity of positive semidefinite matrix factorization. *SIAM Journal on Optimization*, 27(3):1898–1909, 2017. doi:10.1137/16M1080616.

42 Peter W. Shor. Stretchability of Pseudolines is NP-Hard. In Peter Gritzmann and Bernd Sturmfels, editors, *Applied Geometry And Discrete Mathematics*, volume 4 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 531–554, 1991. doi:10.1090/dimacs/004/41.

43 Jeremy Spinrad. Recognition of circle graphs. *Journal of Algorithms. Cognition, Informatics and Logic*, 16(2):264–282, 1994. doi:10.1006/jagm.1994.1012.

44 Jack Stade. Complexity of the boundary-guarding art gallery problem. *CoRR*, abs/2210.12817, 2022. doi:10.48550/arXiv.2210.12817.