arXiv:2302.13597v1 [cs.CG] 27 Feb 2023

The Complexity of Recognizing Geometric Hypergraphs

Daniel Bertschinger[1], Nicolas El Maalouly[1], Linda Kleist[2], Tillmann Miltzow[3], and Simon Weber[1]

[1]Department of Computer Science, ETH Zurich
[2]Department of Computer Science, TU Braunschweig
[3]Department of Information and Computing Sciences, Utrecht University

Abstract

As set systems, hypergraphs are omnipresent and have various representations. In a *geometric representation* of a hypergraph $H = (V, E)$, each vertex $v \in V$ is a associated with a point $p_v \in \mathbb{R}^d$ and each hyperedge $e \in E$ is associated with a connected set $s_e \subset \mathbb{R}^d$ such that $\{p_v \mid v \in V\} \cap s_e = \{p_v \mid v \in e\}$ for all $e \in E$. We say that a given hypergraph H is *representable* by some (infinite) family $\mathcal{F}$ of sets in $\mathbb{R}^d$, if there exist $P \subset \mathbb{R}^d$ and $S \subseteq \mathcal{F}$ such that (P, S) is a geometric representation of H. For a family $\mathcal{F}$, we define RECOGNITION($\mathcal{F}$) as the problem to determine if a given hypergraph is representable by $\mathcal{F}$. It is known that the RECOGNITION problem is $\exists\mathbb{R}$-hard for halfspaces in $\mathbb{R}^d$. We study the families of balls and ellipsoids in $\mathbb{R}^d$, as well as other convex sets, and show that their RECOGNITION problems are also $\exists\mathbb{R}$-complete. This means that these recognition problems are equivalent to deciding whether a multivariate system of polynomial equations with integer coefficients has a real solution.

1 Introduction

As set systems, hypergraphs appear in various contexts, such as databases, clustering, and machine learning. A hypergraph can be represented in various ways. As a generalization of graphs, one can represent vertices by points and hyperedges by connected sets in $\mathbb{R}^d$ such that each set contains exactly the points of a hyperedge. It is desirable that these sets satisfy additional properties, e.g., being (strictly) convex, similar or even translates of each other.

For an introductory example, suppose we are organizing a workshop and have a list of accepted talks. Clearly, each participant wants to quickly identify talks of their specific interest. In order to create a good overview, we want to find a good representation. To this end, we label each talk by several tags, e.g., `hypergraphs`, `graph drawing`, `complexity theory`, `planar graphs`, etc. Then, we create a representation, where each tag is represented by a unit disk (or another nice geometric object of our choice) containing points representing the talks that have this tag, see Figure 1 for an example.. In other words, we are interested in a geometric representation of the hypergraph where the vertex set is given by the talks and tags define the hyperedges.

In this work, we investigate the complexity of deciding whether a given hypergraph has such a geometric representation. We start with a formal definition.

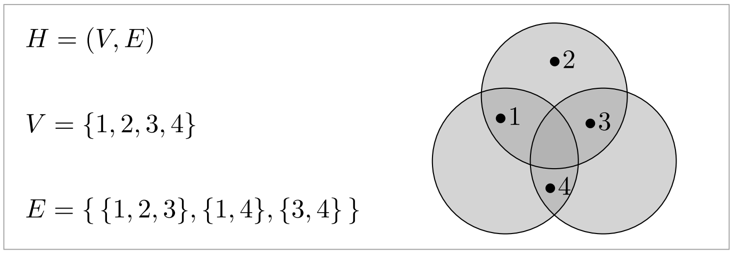

Figure 1: An abstract hypergraph and a geometric representation with unit disks.

Problem Definition. In a *geometric representation* of a hypergraph $H = (V, E)$, each vertex $v \in V$ is associated with a point $p_v \in \mathbb{R}^d$ and each hyperedge $e \in E$ is associated with a connected set $s_e \subset \mathbb{R}^d$ such that $\{p_v \mid v \in V\} \cap s_e = \{p_v \mid v \in e\}$ for all $e \in E$. We say that a given hypergraph H is *representable* by some (possibly infinite) family $\mathcal{F}$ of sets in $\mathbb{R}^d$, if there exist $P \subset \mathbb{R}^d$ and $S \subseteq \mathcal{F}$ such that (P, S) is a geometric representation of H. For a family $\mathcal{F}$ of geometric objects, we define Recognition($\mathcal{F}$) as the problem to determine whether a given hypergraph is representable by $\mathcal{F}$.

Next, we give some definitions describing the geometric families studied in this work.

Bi-curved, Difference-separable, and Computable Convex Sets. We study convex sets that are bi-curved, difference-separable and computable. While the first two properties are needed for $\exists\mathbb{R}$-hardness, the last one is used to show $\exists\mathbb{R}$-membership.

Let $C \subset \mathbb{R}^d$ be a convex set. We call C *computable* if for any point $p \in \mathbb{R}^d$ we can decide on a real RAM whether p is contained in C. We say that C is *bi-curved* if there exists a unit vector $v \in \mathbb{R}^d$, such that there are two distinct tangent hyperplanes on C with normal vector v; with each of these hyperplanes intersecting C in a single point, and C being *smooth* at both of these intersection points. Informally, a convex set is bi-curved, if its boundary has two smoothly curved parts in which the tangent hyperplanes are parallel. Note that a convex, bi-curved set is necessarily bounded. As a matter of fact, any strictly convex bounded set in any dimension is bi-curved. For such sets, any unit vector v fulfills the conditions. As can be seen in Figure 2 (left), being strictly convex is not necessary for being bi-curved.

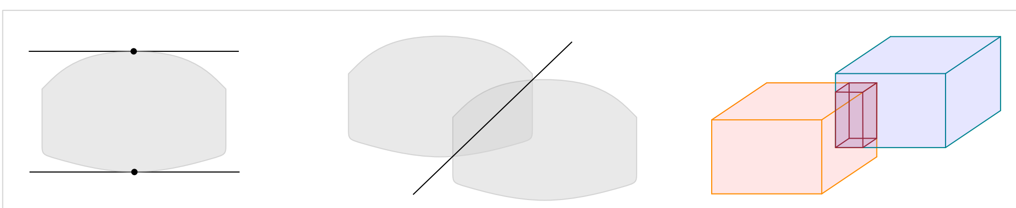

Figure 2: Left: two parallel tangent hyperplanes of a burger-like set proving its bi-curvedness. Middle: a hyperplane separating the symmetric difference of two translates of the burger-like set. Right: two cubes in $\mathbb{R}^3$ whose symmetric difference cannot be separated by a plane.

We call C *difference-separable* if for any two translates C_1, C_2 of C, there exists a hyperplane which strictly separates $C_1 \setminus C_2$ from $C_2 \setminus C_1$. Being difference-separable is fulfilled by any convex set in $\mathbb{R}^2$, see Figure 2 (middle) for an example. For a proof of this fact we refer to [32, Corollary 2.1.2.2]. However, in higher dimensions this is not the case: for a counterexample, consider two 3-cubes as in Figure 2 (right). In higher dimensions, the bi-curved and difference-separable families include the balls and ellipsoids. We are not aware of other natural geometric families with those two properties.

We are now ready to state our results.

Results. Our main contribution is to revive the study of recognition of geometric hypergraphs. We first consider the maybe simplest type of geometric hypergraphs, namely those that stem from halfspaces. It is known due to Tanenbaum, Goodrich, and Scheinerman [57] that the Recognition problem for geometric hypergraphs of halfspaces is NP-hard, but their proof actually implies $\exists\mathbb{R}$-hardness as well. We present a slightly different proof of this fact due to two reasons. Firstly, their proof lacks details about extensions to higher dimensions. Secondly, it is a good stepping stone towards our proof of Theorem 2.

Theorem 1 (Tanenbaum, Goodrich, Scheinerman [57])**.** *For every $d \geq 2$,* Recognition*($\mathcal{F}$) is $\exists\mathbb{R}$-complete for the family $\mathcal{F}$ of halfspaces in $\mathbb{R}^d$.*

Next we consider families of objects that are translates of a given object.

Theorem 2. *For $d \geq 2$, let $C \subseteq \mathbb{R}^d$ be a convex, bi-curved, difference-separable and computable set, and let $\mathcal{F}$ be the family of all translates of C. Then* Recognition*($\mathcal{F}$) is $\exists\mathbb{R}$-complete.*

We note that for $d = 1$, the Recognition problems of halfspaces and translates of convex sets can be solved by sorting and thus can be decided in polynomial time.

One might be under the impression that the Recognition problem is $\exists\mathbb{R}$-complete for every reasonable family of geometric objects of dimension at least two. We show that is not the case by looking at translates of polygons.

Theorem 3. *Let P be a simple polygon with integer coordinates, and $\mathcal{F}$ the family of all translates of P. Then* Recognition*($\mathcal{F}$) is contained in NP.*

Organization. We give an overview over our proof techniques in Section 1.2. Full proofs of Theorem 3 as well as the membership parts of Theorems 1 and 2 are found in Section 2. We introduce the version of pseudohyperplane stretchability used in our hardness reductions in Section 3. Full proofs of the hardness parts of Theorems 1 and 2 can be found in Sections 4 and 5, respectively.

Open problems. As mentioned above, we are not aware of interesting families of bi-curved and difference-separable sets in higher dimensions beyond balls and ellipsoids. The families of translates of a given polygon show the need for some curvature in order to show $\exists\mathbb{R}$-hardness. We wonder if it is sufficient for $\exists\mathbb{R}$-hardness to assume curvature at only one boundary part instead of two opposite ones. Another open question is to consider families that include rotated copies or homothetic copies of a fixed geometric object. Allowing for rotation, it is conceivable that $\exists\mathbb{R}$-hardness even holds for polygons.

1.1 Related work

In this section we give a concise overview over related work on the complexity class $\exists\mathbb{R}$, geometric intersection graphs, and on other set systems related to hypergraphs.

The Existential Theory of the Reals. The complexity class $\exists\mathbb{R}$ (pronounced as 'ER' or 'exists R') is defined via its canonical complete problem ETR (short for *Existential Theory of the Reals*) and contains all problems that polynomial-time many-one reduce to it. In an ETR instance, we are given a sentence of the form

$$\exists x_1, \ldots, x_n \in \mathbb{R} : \varphi(x_1, \ldots, x_n),$$

where φ is a well-formed and quantifier-free formula consisting of polynomial equations and inequalities in the variables and the logical connectives $\{\wedge, \vee, \neg\}$. The goal is to decide whether this sentence is true.

The complexity class $\exists\mathbb{R}$ gains its importance from its numerous influential complete problems. Important $\exists\mathbb{R}$-completeness results include the realizability of abstract order types [40, 52], geometric linkages [45], and the recognition of geometric intersection graphs, as further discussed below. More results concern graph drawing [20, 21, 31, 46], the Hausdorff distance [27], polytopes [19, 43], Nash-equilibria [8, 10, 11, 24, 48], training neural networks [4, 9], matrix factorization [17, 49, 50, 51, 58], continuous constraint satisfaction problems [38], geometric packing [5], the art gallery problem [2, 56], and covering polygons with convex polygons [1].

Geometric Hypergraphs Many aspects of hypergraphs with geometric representations have been studied. Hypergraphs represented by touching polygons in $\mathbb{R}^3$ have been studied by Evans et al. [23]. Bounds on the number of hyperedges in hypergraphs representable by homothets of a fixed convex set S have been established by Axenovich and Ueckerdt [7]. Smorodinsky studied the chromatic number and the complexity of coloring of hypergraphs represented by various types of sets in the plane [54]. Dey and Pach [18] generalize many extremal properties of geometric graphs to hypergraphs where the hyperedges are induced simplices of some point set in $\mathbb{R}^d$. Haussler and Welzl [25] defined ϵ-nets, subsets of vertices of hypergraphs called range spaces with nice properties. Such ϵ-nets of geometric hypergraphs have been studied quite intensely [6, 35, 41, 42].

While there are many structural results, we are not aware of any research into the complexity of recognizing hypergraphs given by geometric representations, other than the recognition of embeddability of simplicial complexes, as we will discuss in the next paragraph.

Other Representations of Hypergraphs. Hypergraphs are in close relation with abstract simplicial complexes. In particular, an abstract simplicial complex (complex for short) is a set system that is closed under taking subsets. A k-complex is a complex in which the maximum size of a set is k. In a geometric representation of an abstract simplicial complex $H = (V, E)$ each ℓ-set of E is represented by a ℓ-simplex such that two simplices of any two sets intersect exactly in the simplex defined by their intersection (and are disjoint in case of an empty intersection). Note that 1-complexes are graphs and hence deciding the representability in the plane corresponds to graph planarity (which is in P). In stark contrast, Abrahamsen, Kleist and Miltzow recently showed that deciding whether a 2-complex has a geometric embedding in $\mathbb{R}^3$ is $\exists\mathbb{R}$-complete [3]; they also prove hardness for other dimensions. Similarly, piecewise linear embeddings of simplicial complexes have been studied [13, 14, 15, 33, 34, 37, 53].

Recognizing Geometric Intersection Graphs. Given a set of geometric objects, its intersection graph has a vertex for each object, and an edge between any two intersecting objects. The complexity of recognizing geometric intersection graphs has been studied for various geometric objects. We summarize these results in Figure 3.

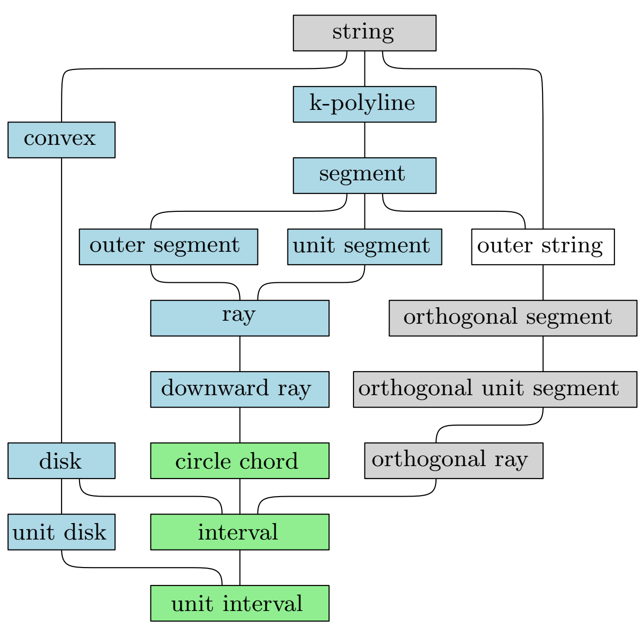

Figure 3: Containment relations of geometric intersection graphs. Recognition of a green class is in P, of a grey class is NP-complete, of a blue class is $\exists\mathbb{R}$-complete, and of a white class is unknown.

While intersection graphs of circle chords (Spinnrad [55]), unit intervals (Looges and Olariu [30]) and intervals (Booth and Lueker [12]) can be recognized in polynomial time, recognizing string graphs (Schaefer and Sedgwick [47]) is NP-complete. In contrast, $\exists\mathbb{R}$-completeness of recognizing intersection graphs has been proved for (unit) disks by McDiarmid and Müller [36], convex sets by Schaefer [44], downward rays by Cardinal et al. [16], outer segments by Cardinal et al. [16], unit segments by Hoffmann et al. [26], segments by Kratochvíl and Matoušek [29], k-polylines by Hoffmann et al. [26], and unit balls by Kang and Müller [28].

The existing research landscape indicates that recognition problems of intersection graphs are $\exists\mathbb{R}$-complete in case that the family of objects satisfy two conditions: Firstly, they need to be "geometrically solid", i.e., not strings. Secondly, some non-linearity must be present by either allowing rotations, or by the objects having some curvature. Our results indicate that this general intuition might translate to the recognition of geometric hypergraphs.

1.2 Overview of Proof Techniques

We prove containment in $\exists\mathbb{R}$ and NP using standard arguments, providing witnesses and verification algorithms.

We prove the hardness parts of Theorems 1 and 2 by reduction from stretchability of pseudohyperplane arrangements. The hypergraph we build from the given arrangement differs from the one built in the proof of Theorem 1 given in [57], since we wish to use a single construction which works nicely for both theorems. Given a simple pseudohyperplane arrangement $\mathcal{A}$, we construct a hypergraph H as follows: We double each pseudohyperplane by giving it a parallel *twin*. In this arrangement, we place a point in every d-dimensional cell. These points represent the vertices of H. Every pseudohyperplane ℓ then defines a hyperedge, which contains all of the points on the same side of ℓ as its twin pseudohyperplane. See Figure 6 for an illustration of this construction.

Because this construction can also be performed on a hyperplane arrangement, it is straightforward to prove that if $\mathcal{A}$ is stretchable, H can be represented by halfspaces. Conversely, we show that the hyperplanes bounding the halfspaces in a representation of H must be a stretching of $\mathcal{A}$.

For Theorem 2, bi-curvedness of a set C implies that locally, C can approximate any halfspace with normal vector close to v as in the definition of bi-curved. This allows us to prove that stretchability of $\mathcal{A}$ implies representability of H by translates of C. The set C being difference-separable is used when reconstructing a hyperplane arrangement from a representation of H.

2 Membership

In this section, we show $\exists\mathbb{R}$- and NP-membership.

Recall that the class NP is usually described by the existence of a witness and a verification algorithm. The same characterization exists for $\exists\mathbb{R}$ using a real verification algorithm. Instead of the witness consisting of binary words of polynomial length, in addition a polynomial number of real-valued numbers are allowed as a witness. Furthermore, in order to be able to use those real numbers, the verification algorithm is allowed to work on the so-called real RAM model of computation. The real RAM allows arithmetic operations with real numbers in constant time [22].

2.1 Halfspaces

Here, we show the $\exists\mathbb{R}$-membership part of Theorem 1.

Lemma 4. *Fix $d \geq 1$ and let $\mathcal{F}$ denote the family of halfspaces in $\mathbb{R}^d$. Then* Recognition*($\mathcal{F}$) is contained in $\exists\mathbb{R}$.*

Proof. We formulate an ETR formula from the hypergraph H as follows. For each vertex/point, we create variables $p = (p_1, \dots, p_d)$ to represent the point. Similarly, for each hyperedge/halfspace, we create variables $h = (h_1, \dots, h_{d+1})$ to represent the coefficients of the halfspace. Then for each point p that is supposed to be in some halfspace h, we create the constraint:

$$h_1 p_1 + \dots h_d p_d \leq h_{d+1}.$$

Similarly, if p is not contained in a halfspace h, we create the constraint:

$$h_1 p_1 + \dots h_d p_d > h_{d+1}.$$

This is a valid ETR sentence that is equivalent to the representability of H. Note that for any fixed dimension d the ETR sentence is of polynomial size. ☐

2.2 Translates of Computable Sets

Here, we show the $\exists\mathbb{R}$-membership part of Theorem 2.

Lemma 5. *For some $d \geq 1$, let $C \subseteq \mathbb{R}^d$ be a computable set and let $\mathcal{F}$ be the family of all translates of C. Then,* Recognition*($\mathcal{F}$) is contained in $\exists\mathbb{R}$.*

Proof. We describe a real verification algorithm as mentioned above. The witness consists of the (real) coordinates of the points representing the vertices and the coefficients of the translation vectors representing the hyperedges. By definition of computable, a verification algorithm can efficiently check if each point is contained in the correct set. ☐

2.3 Translates of Polygons – Proof of Theorem 3

Here, we show Theorem 3, i.e., NP-membership of Recognition of translates of some simple polygon P.

Theorem 3. *Let P be a simple polygon with integer coordinates, and $\mathcal{F}$ the family of all translates of P. Then* Recognition*($\mathcal{F}$) is contained in NP.*

Proof. The proof uses a similar argument to the one used to show that the problem of packing translates of polygons inside a polygon is in NP [5]. For an illustration, consider Figure 4. We first triangulate the convex hull of P, such that each edge of P appears in the triangulation. Then, a representation of a hypergraph H by translates of P gives rise to a certificate as follows: For each pair of a point p and a translate o of P, we specify whether p lies in the convex hull of O, and if it does, in which triangle p lies.

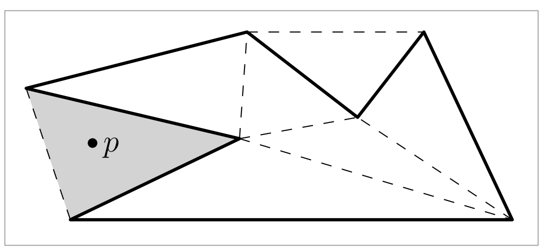

Figure 4: The polygon P, its triangulation, and the triangle that p is contained.

Such a certificate can be tested in polynomial time: we create a linear program whose variables describe the locations of the points p and the translation vectors of each translate of P, and whose constraints enforce the points to lie in the triangles described by the certificate. This linear program has a number of constraints and variables polynomial in the size of H, and can be thus solved in polynomial time.

The solution of this linear program gives the location of the points and the translation vectors of the polygons. This implies that these coordinates are all polynomial and could be used as a certificate directly. ☐

3 Pseudohyperplane Stretchability

A *pseudohyperplane arrangement* in $\mathbb{R}^d$ is an arrangement of *pseudohyperplanes*, where a pseudohyperplane is a set homeomorphic to a hyperplane, and each intersection of pseudohyperplanes is homeomorphic to a plane of some dimension. In the classical definition, every set of d pseudohyperplanes has a non-empty intersection. Here, we consider *partial pseudohyperplane arrangements (PPHAs)*, where not necessarily every set of $\leq d$ pseudohyperplanes has a common intersection.

A PPHA is *simple* if no more than k pseudohyperplanes intersect in a space of dimension $d-k$, in particular, no $d+1$ pseudohyperplanes have a common intersection. We call the 0-dimensional intersection points of d pseudohyperplanes the *vertices* of the arrangement. A simple PPHA $\mathcal{A}$ *stretchable* if there exists a hyperplane arrangement $\mathcal{A}$' such that each vertex in $\mathcal{A}$ also exists in $\mathcal{A}$' and each (pseudo-)hyperplane splits this set of vertices the same way in $\mathcal{A}$ and $\mathcal{A}'$. In other words, each vertex of $\mathcal{A}$ lies on the correct side of each hyperplane in $\mathcal{A}$'. We then call the hyperplane arrangement $\mathcal{A}$' a *stretching* of $\mathcal{A}$.

The problem d-Stretchability is the problem of deciding whether a simple PPHA in $\mathbb{R}^d$ is stretchable. For $d=2$, d-Stretchability contains the stretchability of simple pseudoline arrangements which is known to be $\exists\mathbb{R}$-hard [39, 52]. It is straightforward to prove $\exists\mathbb{R}$-hardness for all $d \geq 2$.

Theorem 6. *d-Stretchability is $\exists\mathbb{R}$-hard for all $d \geq 2$.*

Proof. We reduce from stretchability of simple pseudoline arrangements, which is $\exists\mathbb{R}$-hard as shown in [39, 52].

Consider a simple pseudoline arrangement L in the x_1x_2-plane. We consider $d-2$ pairwise orthogonal hyperplanes $h_1,\ldots,h_{d-2}$ whose common intersection is the x_1x_2-plane; e.g., the hyperplanes defined $x_i = 0$ for $i = 3,\ldots,d$. The intersection of these hyperplanes serves as a canvas in which we aim to embed L. We extend each pseudoline of ℓ to a pseudohyperplane h_ℓ by extending it orthogonally to all $h_1,\ldots,h_{d-2}$, see Figure 5.

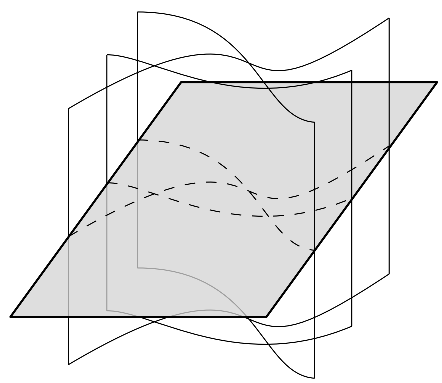

Figure 5: Extending a simple pseudoline arrangement (dashed) to a partial pseudohyperplane arrangement in $\mathbb{R}^3$. The grey hyperplane is the "canvas" hyperplane h_1.

Clearly, the resulting pseudohyperplane arrangement $\mathcal{A}$ can be built in polynomial time. Note that all intersection points of d pseudohyperplanes in $\mathcal{A}$ correspond to intersection points of L.

If L is stretchable, $\mathcal{A}$ is clearly stretchable, as the above construction can be applied to the stretched line arrangement of L.

If $\mathcal{A}$ is stretchable, L is stretchable, since restricting each hyperplane h_ℓ to the intersection of the hyperplanes $h_1,\ldots,h_{d-2}$ yields a line arrangement which is equivalent to L.

As we have thus reduced stretchability of simple pseudoline arrangements to d-Stretchability, this concludes the proof. □

4 Hardness for Halfspaces – Proof of Theorem 1

Proof of Theorem 1. We reduce from d-Stretchability. Let $\mathcal{A}$ be a simple PPHA. For an example consider Figure 6. In a first step, we insert a parallel twin ℓ' for each pseudohyperplane ℓ. The twin is close enough to ℓ such that ℓ and ℓ' have the same intersection pattern. Since ℓ and ℓ' are parallel, they do not intersect each other. This yields an arrangement $\mathcal{A}'$.

In a second step, we introduce a point in each d-dimensional cell of $\mathcal{A}'$; each point represents a vertex in our hypergraph H. Lastly, we define a hyperedge for each pseudohyperplane ℓ of $\mathcal{A}'$: The hyperedge contains all of the points that lie on the same side of the pseudohyperplane as its twin pseudohyperplane. Note that we define a hyperedge for every pseudohyperplane of $\mathcal{A}'$, including the twins inserted in the first step.

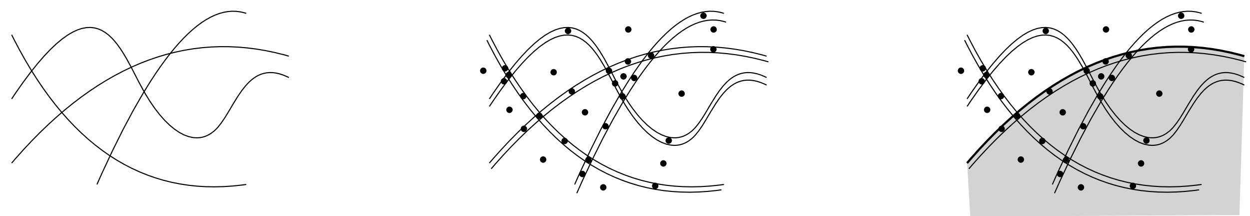

Figure 6: Illustration for the proof of Theorem 1. Construction of the hypergraph H from a simple (partial) pseudohyperplane arrangement $\mathcal{A}$.

It remains to show that H is representable by halfspaces if and only if $\mathcal{A}$ is stretchable. If $\mathcal{A}$ is stretchable, the construction of a representation of H is straightforward: Consider a hyperplane arrangement $\mathcal{B}$ which is a stretching of $\mathcal{A}$. Then, for each hyperplane, we add a parallel hyperplane very close, so that their intersection patterns coincide. This results in a hyperplane arrangement $\mathcal{B}'$. We now prove that every d-dimensional cell of $\mathcal{A}'$ must also exist in $\mathcal{B}'$. First, note that each such cell corresponds to a cell of $\mathcal{A}$, which has at least one vertex on its boundary. All vertices of $\mathcal{A}$ exist in $\mathcal{B}$ by definition of a stretching. Furthermore, the subarrangement of the d hyperplanes in $\mathcal{B}$ intersecting in this vertex must be simple, since their intersection could not be 0-dimensional otherwise. In the twinned hyperplane arrangement $\mathcal{B}'$, all 3^d of the d-dimensional cells incident to this vertex (a cell is given by the following choice for each of the hyperplane pairs: above both hyperplanes, between the hyperplanes, or below both hyperplanes) must exist. This proves that all d-dimensional cells of $\mathcal{A}'$ also exist in $\mathcal{B}'$. Inserting a point in each such d-dimensional cell and considering the (correct) halfspaces bounded by the hyperplanes of $\mathcal{B}'$ yields a representation of H.

We now consider the reverse direction. Let $(P, \mathcal{H})$ be a tuple of points and halfspaces representing H. Let $h_{i,1}$ and $h_{i,2}$ be the two halfspaces associated with a pseudohyperplane ℓ_i of $\mathcal{A}$. Let p_i denote the $(d-1)$-dimensional hyperplane bounding $h_{i,1}$. We show that the family $\{p_i\}_i$ of these hyperplanes is a stretching of $\mathcal{A}$.

For each intersection point q of d pseudohyperplanes $\ell_1, \dots \ell_d$ in $\mathcal{A}$, we consider the corresponding $2d$ pseudohyperplanes in $\mathcal{A}'$. The PPHA $\mathcal{A}'$ contains 3^d d-dimensional cells incident to their 2^d intersections; each of which contains a point. We first show that the associated halfspaces must induce at least 3^d cells, one of which is bounded and represents the intersection point, see also Figure 7: These 3^d points have pairwise distinct patterns of whether or not they are contained in each of the $2d$ halfspaces. Thus, these points need to lie in distinct cells of the arrangement of halfspaces, which proves the claim.

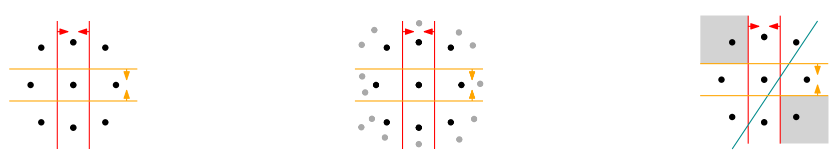

Figure 7: Illustration for the proof of Theorem 1. Representability of H implies stretchability of $\mathcal{A}$.

Moreover, every point in P belongs to exactly one of these 3^d cells. In particular, the central bounded cell, denoted by $c(q)$, contains exactly one point of P.

Now, we argue that the complete cell $c(q)$ (and thus in particular the intersection point of the hyperplanes representing q) lies on the correct side of each hyperplane p in $\{p_i\}_i$. Note that, by construction of the hypergraph H, the 3^d points of q lie on the same side of p. Suppose for a contradiction that p intersects $c(q)$. Then there exist two unbounded cells incident to $c(q)$ which lie on different sides of p; these cells can be identified by translating p until it intersects $c(q)$ only in the boundary. This yields a contradiction to the fact that the 3^d points of q lie on the same side of p.

We conclude that each intersection point of d pseudohyperplanes in $\mathcal{A}$ also exists in the arrangement $\{p_i\}_i$ and lies on the correct side of all hyperplanes. Thus, $\{p_i\}_i$ is a stretching of $\mathcal{A}$ and $\mathcal{A}$ is stretchable. $\square$

5 Hardness for Convex, Bi-curved, and Difference-separable Sets – Proof of Theorem 2

We are now going to prove the hardness part of Theorem 2. To this end, consider any fixed convex, bi-curved, and difference-separable set C in $\mathbb{R}^d$. Note that we can assume C to be fully-dimensional, since otherwise each connected component would live in some lower-dimensional affine subspace, with no interaction between such components. We use the same reduction from the problem d-Stretchability as in the proof for halfspaces in the previous section and show that the constructed hypergraph H is representable by translates of C if and only if the given PPHA $\mathcal{A}$ is stretchable.

Lemma 7. *If $\mathcal{A}$ is stretchable, H is representable by translates of C.*

Proof. We assume that $\mathcal{A}$ is stretchable. We already proved in the previous section that thus there exists an arrangement of hyperplanes, in which we can create a twin of each hyperplane (with a tiny distance α between the twins), and in which we can place all the vertices of H in the appropriate d-dimensional cells. If a vertex is placed between two twin hyperplanes, we assume it to be equidistant to them. As before, we denote this arrangement of hyperplanes and points by $\mathcal{B}'$.

Let v be the unit vector certifying that C is bi-curved; recall the definition in Section 1. Because C is smooth at the touching points of the tangent hyperplanes with normal vector v, there exists $\epsilon > 0$, such that any unit vector w with $\|w - v\|_2 \leq \varepsilon$ also fulfill the conditions to certify that C is bi-curved.

We now assume that $\mathcal{B}'$ fulfills the following properties:

1. the normal vectors of all hyperplanes have distance at most ε to v or to $-v$
2. every intersection point of d hyperplanes as well as every point representing a vertex of H, is contained in $[-1, 1]^d$.

Both properties can be achieved by applying some affine transformation with positive determinant, thus preserving the combinatorial structure of $\mathcal{B}'$.

To represent the hyperedges of H, we will now use very large copies of C. Note that technically we are not allowed to scale C, but scaling C by a factor f is equivalent to scaling the arrangement by a factor $1/f$. Let C^f be the set C scaled by factor f.

In order to determine the necessary scaling factor f, we consider the curvature of C^f in all the points where the tangent hyperplanes of C^f with normal vector w for $\|w - v\|_2 \leq \varepsilon$ intersect C^f. In each such tangent hyperplane h with (unit) normal vector w, we draw a $(d-1)$-ball B of radius $10\sqrt{d}$ around the touching point $h \cap C^f$. Note that $10\sqrt{d}$ is larger than the length of any line segment contained in the box $[-1, 1]^d$. Now, f has to be large enough such that C^f contains every point $p + w \cdot \lambda$, for $p \in B$ and $\alpha/10 \leq \lambda \leq 10\sqrt{d}$. This ensures that the boundary of C^f does not curve away from the tangent hyperplane too quickly, and that C^f is "thick". In other words, C^f locally behaves like an only very slightly curved halfspace. See Figure 8 for an illustration of this requirement on C^f.

We now replace each hyperplane h of the arrangement $\mathcal{B}'$ by a translate C^f_h of C^f, placed such that h is a tangent hyperplane of C^f_h, the single point $h \cap C^f_h$ lies within the box $[-1, 1]^d$, and C^f_h lies completely to the side of h containing its twin hyperplane. It remains to prove that C^f_h contains exactly those points of $\mathcal{B}'$ which are on this side of h. Firstly, C^f_h cannot contain more points, since C^f_h is a subset of the halfspace delimited by h containing its twin hyperplane. Second, we claim that C^f_h contains all these points. To see this, note that within the box $[-1, 1]^d$ containing all points, the boundary of C^f_h is close enough to h that it must contain all points between h and its twin, since these points are located equidistant to the two hyperplanes. Furthermore, all points on the other side of the twin hyperplane are also contained in C^f_h since within the box $[-1, 1]^d$, the boundary $\delta(C^f_h)$ lies completely between h and its twin hyperplane. □

Lemma 8. *If the hypergraph H is representable by translates of C, then $\mathcal{A}$ is stretchable.*

Proof. Assume H is representable. By construction, the two translates $C_{i,r}, C_{i,l}$ of C corresponding to the two hyperedges of each pseudohyperplane ℓ_i must intersect as they contain at least one common point. We call their convex intersection the *lens* of this pseudohyperplane. For each pseudohyperplane ℓ_i of $\mathcal{A}$, we consider some hyperplane p_i which separates $C_{i,r} \setminus C_{i,l}$ from $C_{i,l} \setminus C_{i,r}$. Such a hyperplane exists since C is

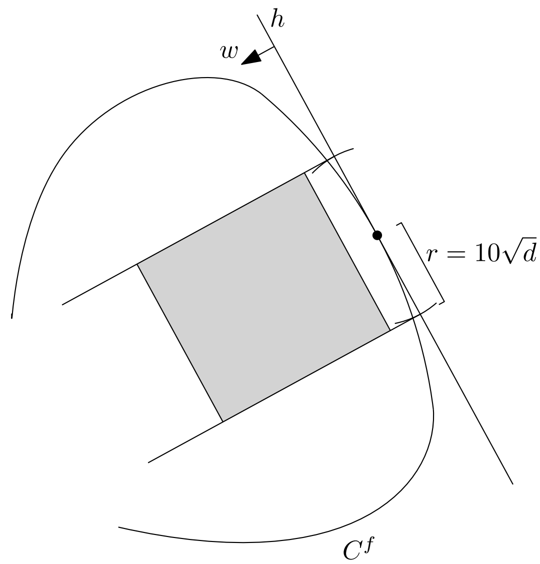

Figure 8: An illustration of the requirement on the scaling factor f. The set C^f must contain the grey region.

difference-separable. Let $\mathcal{P} := \{p_i\}_i$ be the hyperplane arrangement consisting of all these separators. We aim to show that $\mathcal{P}$ is a stretching of $\mathcal{A}$.

To this end, consider d pseudohyperplanes $\ell_1, \ldots, \ell_d$ which intersect in $\mathcal{A}$. For an illustration consider Figure 9. Furthermore, consider one more pseudohyperplane ℓ', and let p', C'_r, C'_l denote the separator hyperplane and translates of C corresponding to ℓ'. We show that the intersection $I_p := p_1 \cap \ldots \cap p_d$ is a single point which lies on the same side of p' as the point $I_\ell := \ell_1 \cap \ldots \cap \ell_d$ lies of ℓ'.

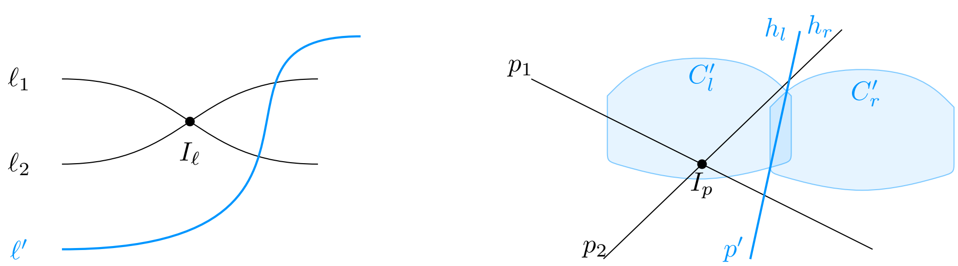

Figure 9: Illustration for the proof of Lemma 8. Left: pseudohyperplanes $\ell_1, \ldots, \ell_d, \ell'$ in $\mathcal{A}$ Right: corresponding hyperplanes $p_1, \ldots, p_d, p'$ in $\mathcal{P}$.

The hyperplane p' divides the space into two halfspaces h_r and h_l such that $C'_r \backslash C'_l \subseteq h_r$ and $C'_l \backslash C'_r \subseteq h_l$. By construction, the two hyperedges defined for ℓ' cover all vertices of H and the vertices in the cells around I_ℓ belong to only one hyperedge. Suppose without loss of generality that these vertices only belong to the hyperedge represented by C'_l. We will show that the intersection I_p must then be a point in h_l.

We first show that the intersection I_p is a point, i.e., 0-dimensional. Consider all 2^d d-dimensional cells of $\mathcal{A}$ around I_ℓ. The construction of H implies that each such cells contains a distinct point, and these points must all lie in distinct cells of the sub-arrangement of the involved hyperplanes $p_1, \ldots, p_d$. Assuming that I_p is not a single point, this sub-arrangement is not simple, and the hyperplanes divide space into strictly fewer than 2^d cells, which results in a contradiction.

Next we prove that I_p is in h_l. Assume towards a contradiction that I_p is in h_r, see also Figure 10. Consider the d lines that are formed by the intersections of subsets of $d-1$ hyperplanes among $p_1, ..., p_d$. Each of these lines is the union of two rays beginning at I_p. Observe that the hyperplane p' can only intersect one of the two rays forming each line. Let S be the convex cone centered at I_p defined by the d

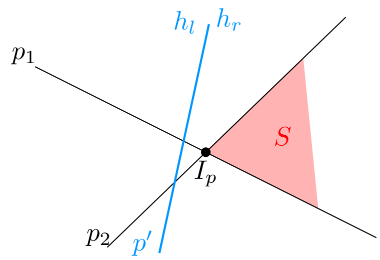

Figure 10: Illustration for the proof of Lemma 8. The cone S must intersect $C'_l \setminus C'_r$, which contradicts I_p lying in h_r.

non-intersected rays. Observe that S does not intersect p', so S must be fully contained in h_r, i.e., $S \cap h_l = \emptyset$. Note, however, by the construction of the hypergraph, there must be a point that lies in $S \cap (C'_l \setminus C'_r) \subseteq S \cap h_l$, which is a contradiction.

We conclude that $\mathcal{P}$ is a stretching of $\mathcal{A}$, and thus $\mathcal{A}$ is stretchable. □

Lemmas 7 and 8 combined now yield hardness of RECOGNITION($\mathcal{F}$) for the family $\mathcal{F}$ of translates of C.

References

[1] 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.

[2] Mikkel Abrahamsen, Anna Adamaszek, and Tillmann Miltzow. The Art Gallery Problem is ∃ℝ-complete. In *STOC 2018: Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 65–73, 2018. doi:10.1145/3188745.3188868.

[3] Mikkel Abrahamsen, Linda Kleist, and Tillmann Miltzow. Geometric embeddability of complexes is ∃r-complete. *CoRR*, abs/2108.02585, 2021. URL: https://arxiv.org/abs/2108.02585, arXiv:2108.02585.

[4] Mikkel Abrahamsen, Linda Kleist, and Tillmann Miltzow. Training Neural Networks is ER-complete. In Marc A. Ranzato, Alina Beygelzimer, K. Nguyen, Percy Liang, Jennifer W. Vaughan, and Yann Dauphin, editors, *Advances in Neural Information Processing Systems (NeurIPS 2021)*, volume 34, 2021.

[5] Mikkel Abrahamsen, Tillmann Miltzow, and Nadja Seiferth. Framework for ER-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] Boris Aronov, Esther Ezra, and Micha Sharir. Small-size ϵ-nets for axis-parallel rectangles and boxes. *SIAM Journal on Computing*, 39(7):3248–3282, 2010. doi:10.1137/090762968.

[7] Maria Axenovich and Torsten Ueckerdt. Density of range capturing hypergraphs. *Journal of Computational Geometry*, 7(1), 2016. doi:10.20382/jocg.v7i1a1.

[8] Marie L. T. Berthelsen and Kristoffer A. Hansen. On the Computational Complexity of Decision Problems About Multi-player Nash Equilibria. In Dimitris Fotakis and Evangelos Markakis, editors, *International Symposium on Algorithmic Game Theory*, volume 11801 of *Lecture Notes in Computer Science*, pages 153–167, 2019. doi:10.1007/978-3-030-30473-7_11.

[9] Daniel Bertschinger, Christoph Hertrich, Paul Jungeblut, Tillmann Miltzow, and Simon Weber. Training fully connected neural networks is ∃ℝ-complete. *arXiv preprint arXiv:2204.01368*, 2022.

[10] Vittorio Bilò and Marios Mavronicolas. A Catalog of EXISTS-R-Complete Decision Problems About Nash Equilibria in Multi-Player Games. In Nicolas Ollinger and Heribert Vollmer, editors, *33rd Symposium on Theoretical Aspects of Computer Science (STACS 2016)*, Leibniz International Proceedings in Informatics (LIPIcs), pages 17:1–17:13, 2016. `doi:10.4230/LIPIcs.STACS.2016.17`.

[11] Vittorio Bilò and Marios Mavronicolas. Existential-R-Complete Decision Problems about Symmetric Nash Equilibria in Symmetric Multi-Player Games. In Vollmer Heribert and Brigitte Vallée, editors, *34th Symposium on Theoretical Aspects of Computer Science (STACS 2017)*, volume 66 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 13:1–13:14, 2017. `doi:10.4230/LIPIcs.STACS.2017.13`.

[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] Martin Čadek, Marek Krčál, Jiří Matoušek, Francis Sergeraert, Lukáš Vokřínek, and Uli Wagner. Computing all maps into a sphere. *Journal of the ACM*, 61(3):1–44, 2014. `doi:10.1145/2597629`.

[14] Martin Čadek, Marek Krčál, Jiří Matoušek, Lukáš Vokřínek, and Uli Wagner. Time computation of homotopy groups and Postnikov systems in fixed dimension. *SIAM Journal on Computing*, 43(5):1728–1780, 2014. `doi:10.1137/120899029`.

[15] Martin Čadek, Marek Krčál, and Lukáš Vokřínek. Algorithmic solvability of the lifting-extension problem. *DCG*, 57(4):915–965, 2017. `doi:10.1007/s00454-016-9855-6`.

[16] 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`.

[17] 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, 2016. `doi:10.4230/LIPIcs.ICALP.2016.103`.

[18] T. K. Dey and J. Pach. Extremal problems for geometric hypergraphs. *Discrete & Computational Geometry*, 19(4):473–484, Apr 1998. `doi:10.1007/PL00009365`.

[19] Michael G. Dobbins, Andreas Holmsen, and Tillmann Miltzow. A Universality Theorem for Nested Polytopes. arXiv preprint, 2019. `arXiv:1908.02213`.

[20] Michael G. Dobbins, Linda Kleist, Tillmann Miltzow, and Paweł Rzążewski. Completeness for the complexity class $\forall\exists\mathbb{R}$ and area-universality. *Discrete & Computational Geometry (DCG)*, 2022. `doi:10.1007/s00454-022-00381-0`.

[21] Jeff Erickson. Optimal Curve Straightening is $\exists\mathbb{R}$-Complete. arXiv preprint, 2019. `arXiv:arXiv:1908.09400`.

[22] 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`.

[23] William Evans, Paweł Rzążewski, Noushin Saeedi, Chan-Su Shin, and Alexander Wolff. Representing graphs and hypergraphs by touching polygons in 3d. In Daniel Archambault and Csaba D. Tóth, editors, *Graph Drawing and Network Visualization*, pages 18–32, Cham, 2019. Springer International Publishing. `doi:10.1007/978-3-030-35802-0_2`.

[24] Jugal Garg, Ruta Mehta, Vijay V. Vazirani, and Sadra Yazdanbod. $\exists\mathbb{R}$-Completeness for Decision Versions of Multi-Player (Symmetric) Nash Equilibria. *ACM Transactions on Economics and Computation*, 6(1):1:1–1:23, 2018. `doi:10.1145/3175494`.

[25] David Haussler and Emo Welzl. Epsilon-nets and simplex range queries. In *Proceedings of the second annual symposium on Computational geometry*, pages 61–71, 1986.

[26] Michael Hoffmann, Tillmann Miltzow, Simon Weber, and Lasse Wulf. Recognition of unit segment graphs is $\exists\mathbb{R}$-complete. Unpublished, in preparation.

[27] Paul Jungeblut, Linda Kleist, and Tillmann Miltzow. The complexity of the hausdorff distance. In Xavier Goaoc and Michael Kerber, editors, *38th International Symposium on Computational Geometry, SoCG 2022, June 7-10, 2022, Berlin, Germany*, volume 224 of *LIPIcs*, pages 48:1–48:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. `doi:10.4230/LIPIcs.SoCG.2022.48`.

[28] Ross J. Kang and Tobias Müller. Sphere and dot product representations of graphs. *Discrete & Computational Geometry*, 47(3):548–568, Apr 2012. `doi:10.1007/s00454-012-9394-8`.

[29] 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`.

[30] Peter J. Looges and Stephan Olariu. Optimal greedy algorithms for indifference graphs. *Computers & Mathematics with Applications*, 25(7):15–25, 1993. `doi:10.1016/0898-1221(93)90308-I`.

[31] Anna Lubiw, Tillmann Miltzow, and Debajyoti Mondal. The Complexity of Drawing a Graph in a Polygonal Region. In Therese Biedl and Andreas Kerren, editors, *GD 2018: Graph Drawing and Network Visualization*, volume 11282 of *Lecture Notes in Computer Science*, pages 387–401, 2018. `doi:10.1007/978-3-030-04414-5_28`.

[32] Lihong Ma. *Bisectors and Voronoi Diagrams for Convex Distance Functions*. PhD thesis, FernUniversität Hagen, 2000.

[33] Jiří Matoušek, Eric Sedgwick, Martin Tancer, and Uli Wagner. Embeddability in the 3-sphere is decidable. *Journal of the ACM*, 65(1):1–49, 2018. `doi:10.1145/2582112.2582137`.

[34] Jiří Matoušek, Martin Tancer, and Uli Wagner. Hardness of embedding simplicial complexes in $\mathbb{R}^d$. *JEMS*, 13(2):259–295, 2011. `doi:10.4171/JEMS/252`.

[35] Jiří Matoušek, Raimund Seidel, and E. Welzl. How to net a lot with little: Small ϵ-nets for disks and halfspaces. In *Proceedings of the Sixth Annual Symposium on Computational Geometry*, SCG '90, page 16–22, New York, NY, USA, 1990. Association for Computing Machinery. `doi:10.1145/98524.98530`.

[36] Colin McDiarmid and Tobias Müller. Integer realizations of disk and segment graphs. *Journal of Combinatorial Theory, Series B*, 103(1):114–143, 2013.

[37] Arnaud de Mesmay, Yo'av Rieck, Eric Sedgwick, and Martin Tancer. Embeddability in $\mathbb{R}^3$ is NP-hard. *Journal of the ACM*, 67(4):20:1–20:29, 2020. `doi:10.1145/3396593`.

[38] Tillmann Miltzow and Reinier F. Schmiermann. On Classifying Continuous Constraint Satisfaction Problems. In Nisheeth K. Vishnoi, editor, *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 781–791, 2022. `doi:10.1109/FOCS52979.2021.00081`.

[39] N. E. Mnev. *The universality theorems on the classification problem of configuration varieties and convex polytopes varieties*, pages 527–543. Springer Berlin Heidelberg, Berlin, Heidelberg, 1988. `doi:10.1007/BFb0082792`.

[40] Nikolai E. Mnëv. The Universality Theorems on the Classification Problem of Configuration Varieties and Convex Polytopes Varieties. In Oleg Y. Viro and Anatoly M Vershik, editors, *Topology and Geometry — Rohlin Seminar*, volume 1346 of *Lecture Notes in Mathematics*, pages 527–543. Springer, 1988. `doi:10.1007/BFb0082792`.

[41] János Pach and Gábor Tardos. Tight lower bounds for the size of epsilon-nets. *Journal of the American Mathematical Society*, 26(3):645–658, 2013.

[42] János Pach and Gerhard Woeginger. Some new bounds for epsilon-nets. In *Proceedings of the Sixth Annual Symposium on Computational Geometry*, SCG '90, page 10–15, New York, NY, USA, 1990. Association for Computing Machinery. doi:10.1145/98524.98529.

[43] 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. doi:10.1090/S0273-0979-1995-00604-X.

[44] 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.

[45] 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.

[46] 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.

[47] Marcus Schaefer, Eric Sedgwick, and Daniel Štefankovič. Recognizing string graphs in np. *Journal of Computer and System Sciences*, 67(2):365–380, 2003. Also STOC 2002.

[48] Marcus Schaefer and Daniel Štefankovič. Fixed Points, Nash Equilibria, and the Existential Theory of the Reals. *Theory of Computing Systems*, 60:172–193, 2017. doi:10.1007/s00224-015-9662-0.

[49] 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.

[50] Yaroslav Shitov. A Universality Theorem for Nonnegative Matrix Factorizations. arXiv preprint, 2016. arXiv:1606.09068.

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

[52] 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.

[53] Arkadiy Skopenkov. Extendability of simplicial maps is undecidable. arXiv preprint, 2020. arXiv:2008.00492.

[54] Shakhar Smorodinsky. On the chromatic number of geometric hypergraphs. *SIAM Journal of Discrete Mathematics*, 21(3):676–687, 2007. doi:10.1137/050642368.

[55] Jeremy Spinrad. Recognition of circle graphs. *Journal of Algorithms*, 16(2):264–282, 1994.

[56] Jack Stade. Complexity of the boundary-guarding art gallery problem. *arXiv preprint arXiv:2210.12817*, 2022.

[57] Paul J. Tanenbaum, Michael T. Goodrich, and Edward R. Scheinerman. Characterization and recognition of point-halfspace and related orders. In Roberto Tamassia and Ioannis G. Tollis, editors, *Graph Drawing*, pages 234–245, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[58] Levent Tuncel, Stephen Vavasis, and Jingye Xu. Computational complexity of decomposing a symmetric matrix as a sum of positive semidefinite and diagonal matrices. *arXiv preprint arXiv:2209.05678*, 2022.