# Hypergraph Representation via Axis-Aligned Point-Subspace Cover

**Oksana Firman[1] and Joachim Spoerhase[1]**

1 **Universität Würzburg, Germany**
`firstname.lastname@uni-wuerzburg.de`

## Abstract

We propose a new representation of $k$-partite, $k$-uniform hypergraphs (which we call $k$-hypergraphs for short) by a finite set $P$ of points in $\mathbb{R}^d$ and a parameter $\ell \leq d-1$. Each point in $P$ is covered by $k = \binom{d}{\ell}$ many axis-aligned affine $\ell$-dimensional subspaces of $\mathbb{R}^d$, which we call $\ell$-subspaces for brevity. We interpret each point in $P$ as a hyperedge that contains each of the covering $\ell$-subspaces as a vertex. The class of $(d, \ell)$-*hypergraphs* is the class of $k$-hypergraphs that can be represented in this way, where $k = \binom{d}{\ell}$. The resulting classes of hypergraphs are fairly rich: Every $k$-hypergraph is a $(k, k-1)$-hypergraph. On the other hand, $(d, \ell)$-hypergraphs form a proper subclass of the class of all $\binom{d}{\ell}$-hypergraphs for $\ell < d-1$.

We are able to give a natural structural characterization of $(d, \ell)$-hypergraphs based on vertex cuts. This characterization leads to a polynomial-time recognition algorithm that decides for a given $\binom{d}{\ell}$-hypergraph whether or not it is a $(d, \ell)$-hypergraph and that computes a representation if existing. We assume that the dimension $d$ is constant and that the partitioning of the vertex set is prescribed. For the sake of presentation, we describe in this paper our result for the case of $(d, 1)$-hypergraphs, that is, for covering points with axis-parallel lines. This special case can be naturally extended to the general case.

## 1 Introduction

**Motivation and Related Work.** Geometric representations of graphs or hypergraphs is wide and intensively studied field of research. Well-known examples are geometric intersection or incidence graphs with a large body of literature [12, 5, 13]. The benefit of studying geometric graph representations is two-fold. On the one hand, knowing that a given graph can be represented geometrically may give new insights because the geometric perspective is often more intuitive. On the other hand, giving a graphical characterization for certain types of geometric objects may help pin down the essential combinatorial properties that can be exploited in the geometric setting.

One example of this interplay is the study of geometric set cover and geometric hitting set problems in geometric optimization [2, 14, 3]. In this important branch of computational geometry, incidence relations of two types of geometric objects are studied where one object type is represented by vertices of a hypergraph whose hyperedges are, in turn, represented by the other object type. In this representation a vertex is contained in a hyperedge if and only if the corresponding geometric objects have a certain geometric relation such as containment or intersection. The objective is to find the minimum number of nodes hitting all hyperedges[1]. In this line of research, the goal is to exploit the geometry in order to improve upon the state of the art for general hypergraphs. This is known to be surprisingly challenging even in many seemingly elementary settings.

[1] For the sake of presentation, we use here the representation as hitting set problem rather than the equivalent and maybe more common geometric set cover interpretation.

37th European Workshop on Computational Geometry, St. Petersburg, Russia, April 7–9, 2021.

For example, in the well-studied point line cover problem [8, 10] we are given a set of points in the plane and a set of lines. The goal is to identify a smallest subset of the lines to cover all the points. This problem can be cast as a hypergraph vertex cover problem. Points can be viewed as hyperedges containing the incident lines as vertices. The objective is to cover all the hyperedges by the smallest number of vertices.

It seems quite clear that point line cover instances form a heavily restricted subclass of general hypergraph vertex cover. For example, they have the natural intersection property that two lines can intersect in at most one point. However, somewhat surprisingly, in terms of approximation algorithms, no worst-case result improving the ratios for general hypergraph vertex cover [1, 7, 4] is known. In fact, it has been shown that merely exploiting the above intersection property in the hypergraph vertex cover is not sufficient to give improved approximations [11]. Giving a simple combinatorial characterization of the point line cover instances seems to be an intriguing task.

In this paper, we study a representation of hypergraphs that arises from a natural variant of point line cover where we want to cover a given set $P$ of points in $\mathbb{R}^d$ by *axis-parallel* lines. While the axis-parallel case of point line cover has been considered before [6] the known algorithms do not improve upon the general case of hypergraph vertex cover [1, 4]. More generally, we investigate the generalization where we are additionally given a parameter $\ell \leq d-1$ and we would like to cover $P$ by axis-aligned affine $\ell$-dimensional subspaces of $\mathbb{R}^d$, which we call $\ell$-subspaces. The resulting classes of hypergraphs is fairly rich as any $k$-partite $k$-uniform hypergraph (i.e. a hypergraph with a partition of vertices into $k$ parts such that each hyperedge contains at most one vertex of each type and together exactly $k$ vertices) can be represented by a set of points in $\mathbb{R}^k$ to be covered by $(k-1)$-subspaces. On the other hand for $\ell < d-1$ we obtain proper subclasses of all $k$-hypergraphs.

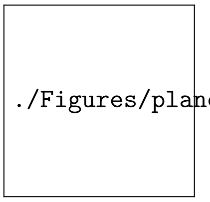
We remark that our representation does not exploit the geometry of the Euclidean $\mathbb{R}^d$. Rather, the representation can also be considered on a hypercube $X^d$ for some set $X$ where subspaces are subsets of $X^d$ fixing certain coordinates. We feel that the usage of the geometric language is more intuitive.

**Our Contribution** To the best of our knowledge, we are the first to study the representation of $k$-hypergraphs via axis-aligned point subspace cover instance in this generality. Our main insight is that the axis-aligned case of point subspace cover allows for a natural, combinatorial characterization contrasting what is known for the non-aligned case. The characterization is based on vertex cuts and can be leveraged to obtain a polynomial time recognition algorithm for such hypergraphs assuming the dimension $d$ is a constant and that we are given the partition of the vertices (which is NP-hard to compute for $k \geq 3$ [9]). We believe that it is an interesting research direction to exploit these combinatorial properties in order to obtain improved results for various optimization problems in $(d, \ell)$-hypergraphs such as hypergraph vertex cover or hypergraph matching.

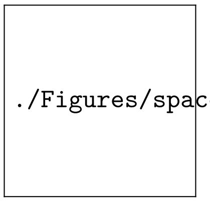
## 2 Point Line Cover and Hypergraph Representation

For the sake of an easier presentation, we describe the result for the special case of point line covers. We give later some intuition how to generalize the results to high-dimensional axis-aligned affine subspaces.

Let $P$ be a finite set of points in $\mathbb{R}^d$. Then we define the hypergraph $G_P$ as follows. The vertex set of $G_P$ is the set of axis-parallel lines containing at least one point in $P$. The hyperedges in $G_P$ correspond to the points in $P$ where the hyperedge corresponding to some




(a)




(b)

**Figure 1** A graph (a) and a hypergraph (b) and their representations in 2D and 3D, respectively.

$p \in P$ contains the $k$ axis-parallel lines incident on $p$ as vertices. Note that $G_P$ is $k$-partite and $k$-uniform (that is $G_P$ is a $k$-hypergraph) where the $k$ groups of the partition correspond to the $k$ dimensions.

Our main task is to decide for a given hypergraph $G$ whether there is a point line cover instance $P$ such that $G$ and $G_P$ are isomorphic. We say that $G$ is *represented* by $P$ and, thus, *representable*. We assume that the partition of $G$ into $k$ groups is given.

More formally, we want to compute for a given $k$-hypergraph $G = (V_1 \cup V_2 \cup, \ldots, \cup V_k, E)$ a point line cover instance $P$ such that each $e = (v_1, \ldots, v_k) \in E$ corresponds to some $p^e = (x_1^e, \ldots, x_k^e) \in P$ and where $v_i \in V_i$ corresponds to the line $l^{v_i}$ that is parallel to the $i$th coordinate axis and contains $p_e$, that is, for all $j \neq i$, we fix the coordinates $x_j^e, j \neq i$ whereas the $i$th coordinate is free, see Fig. 1 for examples.

We remark that every bipartite graph is representable in $\mathbb{R}^2$ because any adjacency matrix can be represented by a grid-like construction as shown in Fig. 1a. Therefore, from now on we consider the case $k \geq 3$.

## 3 Characterization of Representable Hypergraphs

We use the notation $[k] = \{1, \ldots, k\}$ for $k \in \mathbb{N}$. Let $G = (V = V_1 \cup \cdots \cup V_k, E)$ be a $k$-hypergraph.

▶ **Definition 3.1.** Let $s, t \in V$. An *$s$–$t$ path* is a sequence of vertices $s = v_1, \ldots, v_r = t$ such that $v_i$ and $v_{i+1}$ are both contained in some hyperedge $e \in E$ for all $i \in [r-1]$. Similarly, if $e, e' \in E$ then an *$e$–$e'$ path* is a $v$–$v'$ path such that $v \in e$ and $v' \in e'$.

▶ **Definition 3.2** (Vertex separability)**.** For a given $k$-hypergraph $G$ two distinct vertices $v$ and $v'$ from the same group $V_i$, $i \in [k]$ are *separable* if there exists some $j \in [k]$ with $j \neq i$ such that every $v$–$v'$ path contains a vertex in $V_j$. (Informally, removing $V_j$ from the vertex set and from the edges separates $v$ and $v'$.) A hypergraph is called *vertex-separable* if every two vertices from the same group are separable.

we were not consistent with the hyphen in vertex separability and edge separability. Smth to change in the full version.

▶ **Definition 3.3** (Edge separability)**.** For a given $k$-hypergraph $G$ two distinct hyperedges $e$ and $e'$ are *separable* if there exists some $j \in [k]$ such that every $e$–$e'$ path contains a vertex in $V_j$. A hypergraph is called *edge separable* if every two hyperedges are separable.

Note that any pair of hyperedges sharing two or more vertices are not separable. Therefore, edge-separable hypergraphs do not contain such hyperedge pairs.

▶ **Lemma 3.4.** *Vertex separability implies edge separability.*

**Proof.** Assume that a given $k$-hypergraph $G$ is not edge-separable. This means that there are two distinct hyperedges $e$ and $e'$ that are not separable. Then $\forall j \in [k]$ there is an $e$–$e'$

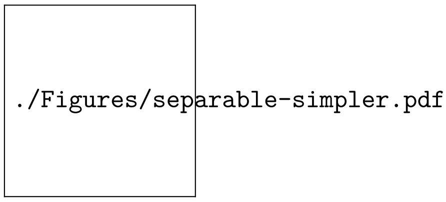



**Figure 2** A hypergraph (on the left) that is edge-separable, but not vertex-separable (the vertices from $V_2$ are not separable). The line $\ell^{v'}$ (on the right) must simultaneously intersect $\ell^{u}$ and $\ell^{u'}$ and therefore must be equal to $\ell^{v}$. A contradiction.

path that does not contain a vertex from $V_j$. Because $e$ and $e'$ are distinct there are distinct vertices $v$ and $v'$ with $v \in e$ and $v' \in e'$ from the same group $V_i$ for some $i \in [k]$. Now, for each $j \in [k]$, $j \neq i$ there exists an $e$–$e'$ path $P_j$ that does not contain any vertex from $V_j$. But then $v, P_j, v'$ forms a $v$–$v'$ path not containing any vertex from $V_j$. This means that $G$ is not vertex-separable. ◀

The converse is not true, see Fig. 2. In the instance depicted, the two red edges, for example, are separated by removing the orange vertex part $V_1$ and the two black edges are separated by removing the green vertex part $V_3$.

▶ **Definition 3.5.** Let $G$ be a $k$-hypergraph. For each $i \in [k]$ we construct a graph $G_i = (E, E_i)$ as follows: $e$ and $e' \in E$ are adjacent if and only if $e$ and $e'$ have a common vertex in a group $V_j$ with $j \neq i$.

change $l$ to $\ell$ in the proof

▶ **Theorem 3.6.** *A $k$-hypergraph $G$ is representable if and only if it is vertex-separable.*

**Proof.** We construct for each hyperedge $e$ a point $p^e \in \mathbb{R}^k$ and for each vertex $v_i \in V_i$ with $i \in [k]$ a line $l^{v_i} \subseteq \mathbb{R}^k$ that is parallel to the $x_i$-axis. We do this as follows. For $G$ we construct the graphs $G_i$, $i \in [k]$. For each graph $G_i$ we consider the connected components of the graph and assign to each of them a unique (integer) value.

Now, if $p^e_i$ is the value of the connected component in $G_i$ that contains $e$ then we let the point $p^e = (p^e_1, \ldots, p^e_k)$ represent the hyperedge $e$, see Fig. 3 for an example.

Recall that any line parallel to the $x_i$-axis can be defined by fixing its $x_j$-coordinate for all $j \neq i$, while leaving $x_i$ free. Now, if the hyperedge $e = \{v_1, \ldots, v_k\}$ is represented by $p^e = (p^e_1, \ldots, p^e_k)$ then for each $i \in [k]$, the line $l^{v_i}$ that represents the vertex $v_i$ is defined by coordinates $p^e_j$, $j \neq i$ while leaving the $x_i$-coordinate free, see Fig. 3. It is important to note, that the representation $l^{v_i}$ is well-defined although $v_i$ may be contained in multiple hyperedges in $G$. This follows from the fact that all the hyperedges containing $v_i$ belong to the same connected component in $G_j$, $j \in [k]$, $j \neq i$ because each pair of them is joined by some edge in $G_j$ corresponding to $v_i$ and in particular these hyperedges form a clique. Therefore, there is no disagreement in the $x_j$-coordinate where $j \neq i$. Hence, we uniquely define the coordinates that determine a line.

($\Leftarrow$) Assume that $G$ is vertex-separable. By the construction of the point line cover instance we have:

- every point $p^e$ is in fact covered by the lines $l^{v_1}, \ldots, l^{v_k}$ where $e = \{v_1, \ldots, v_k\}$, because by construction every line $l^{v_i}$ and point $p^e$ have the same $x_j$-coordinate with $j \neq i$.
- $\forall v \neq v' \in V$ it holds $l^v \neq l^{v'}$. This is obviously true if vertices belong to different groups, because then the free coordinate of $v$ is fixed for $v'$ and vice versa. If $v, v' \in V_i$ for some $i \in [k]$ then, by vertex separability, there exists $j \neq i$ such that $v$ and $v'$ are not connected in graph $G_j$ and get different $x_j$-coordinates. So they represent distinct lines.

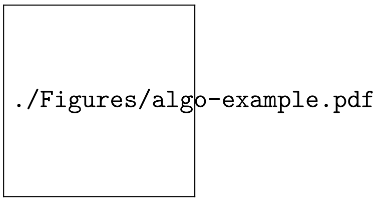



**Figure 3** The graphs $G_1, G_2, G_3$ and the coordinates of the points and lines corresponding to the hyperedges and vertices. The dots instead of coordinates mean that those coordinates are free.

- $\forall e \neq e' \in E$ it holds $p^e \neq p^{e'}$. Indeed, by Lemma 3.4, $G$ is edge-separable and by definition of edge separability distinct hyperedges are not connected in at least one graph $G_i$ and get different $x_i$-coordinates. So they represent distinct points.

By the above construction, for every incident vertex-hyperedge pair $v \in V, e \in E$, that is, $v \in e$, the corresponding geometric objects $l^v$ and $p^e$ are incident as well. We claim that if $v$ and $e$ are not incident, that is, $v \notin e$ then $l^v$ and $p^e$ are not incident as well. This is because every point $p^e$ is already incident to precisely $k$ lines $l^v$ by construction, because the lines $l^v$ are pairwise distinct, and because $p_e$ cannot be incident on more than $k$ axis-parallel lines. Thus we constructed a point line cover instance that represents the hypergraph $G$ and this means that $G$ is representable.

($\Rightarrow$) Assume that $G$ is not vertex-separable but that it has a point line cover representation. This means that it contains at least two distinct vertices $v$ and $v'$ from the same group $V_i$ that are not separable. Then for each group $V_j$ with $j \neq i$, there exists a $v$–$v'$ path $v = v_1, \ldots, v_r = v'$ such that $v_t \notin V_j$ for each $t \in [r]$. All lines $l^{v_t}$ with $t \in [r]$ that represent the vertices $v_1, \ldots, v_r$ lie on the same hyperplane $H_j$ perpendicular to the $x_j$-axis. This is because successive line pairs are joined by a common point (representing the hyperedge containing both) and since none of these lines is parallel to the $x_j$-axis and so the $x_j$-coordinate stays fixed. Since this holds for all $j \in [k]$, $j \neq i$, the lines $l^v$ and $l^{v'}$ lie in the intersection $\bigcap_{j \neq i} H_j$. But the intersection of such hyperplanes is a single line. This contradicts that $v$ and $v'$ correspond to the distinct lines. ◀

## 4 Further Results and Open Questions.

In the full version of the paper, we also present a polynomial time algorithm to compute for a vertex separable hypergraph a point line cover representation. The algorithm implements the idea used in Theorem 3.6. We are also able to generalize the result from point line covers to point subspace covers. That is, we provide a characterization of the more general case of $(d, \ell)$-hypergraphs along with a recognition algorithm for constant $d$.

We conclude with some open questions. Can we leverage our combinatorial characterizations to give improved algorithms for classical optimization problems such as hypergraph vertex cover or hypergraph matching for $(d, \ell)$-hypergraphs? What is the relation of such graphs to other more well-studied graph classes? Is there a polynomial recognition algorithm also for non-constant $d$? What about point line cover in the plane with a fixed number of possible directions of the lines?

## References


1 Ron Aharoni, Ron Holzman, and Michael Krivelevich. On a theorem of lovász on covers in tau-partite hypergraphs. *Comb.*, 16(2):149–174, 1996. doi:10.1007/BF01844843.

2 Hervé Brönnimann and Michael T. Goodrich. Almost optimal set covers in finite vc-dimension. *Discret. Comput. Geom.*, 14(4):463–479, 1995. doi:10.1007/BF02570718.

3 Timothy M. Chan, Elyot Grant, Jochen Könemann, and Malcolm Sharpe. Weighted capacitated, priority, and geometric set cover via improved quasi-uniform sampling. In Yuval Rabani, editor, *Proc. Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'12)*, pages 1576–1585. SIAM, 2012. doi:10.1137/1.9781611973099.125.

4 Vasek Chvátal. A greedy heuristic for the set-covering problem. *Math. Oper. Res.*, 4(3):233–235, 1979. doi:10.1287/moor.4.3.233.

5 H. S. M. Coxeter. Self-dual configurations and regular graphs. *Bull. Amer. Math. Soc.*, 56(5):413–455, 09 1950.

6 Daya Ram Gaur and Binay Bhattacharya. Covering points by axis parallel lines. In *Proc. 23rd European Workshop on Computational Geometry (EuroCG'07)*, pages 42–45. Citeseer, 2007.

7 Venkatesan Guruswami and Rishi Saket. On the inapproximability of vertex cover on $k$-partite $k$-uniform hypergraphs. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide, and Paul G. Spirakis, editors, *Proc. 37th International Colloquium on Automata, Languages and Programming (ICALP'10)*, volume 6198 of *LNCS*, pages 360–371. Springer, 2010. doi:10.1007/978-3-642-14165-2_31.

8 Refael Hassin and Nimrod Megiddo. Approximation algorithms for hitting objects with straight lines. *Discret. Appl. Math.*, 30(1):29–42, 1991. doi:10.1016/0166-218X(91)90011-K.

9 Lucian Ilie, Roberto Solis-Oba, and Sheng Yu. Reducing the size of NFAs by using equivalences and preorders. In Alberto Apostolico, Maxime Crochemore, and Kunsoo Park, editors, *Combinatorial Pattern Matching*, pages 310–321. SV, 2005. doi:10.1007/11496656_27.

10 Stefan Kratsch, Geevarghese Philip, and Saurabh Ray. Point line cover: The easy kernel is essentially tight. In Chandra Chekuri, editor, *Proc. 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'14)*, pages 1596–1606. SIAM, 2014. doi:10.1137/1.9781611973402.116.

11 V. S. Anil Kumar, Sunil Arya, and H. Ramesh. Hardness of set cover with intersection 1. In Ugo Montanari, José D. P. Rolim, and Emo Welzl, editors, *Proc. International Colloquium on Automata, Languages and Programming (ICALP'00)*, volume 1853 of *LNCS*, pages 624–635. Springer, 2000. doi:10.1007/3-540-45022-X_53.

12 Terry A. McKee and F. R. McMorris. *Topics in Intersection Graph Theory.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1999.

13 Tomaž Pisanski and Milan Randić. Bridges between geometry and graph theory. In *in Geometry at Work, C.A. Gorini, ed., MAA Notes 53*, pages 174–194. America.

14 Kasturi R. Varadarajan. Weighted geometric set cover via quasi-uniform sampling. In Leonard J. Schulman, editor, *Proc. 42nd ACM Symposium on Theory of Computing (STOC'10)*, pages 641–648. ACM, 2010. doi:10.1145/1806689.1806777.