Decomposition-width: Extending the Clique-width to Hypergraphs

Yann Strozecki

Université Paris-Sud 11, Laboratoire de Recherche en Informatique.

Abstract. We define a width parameter for hypergraphs, which we call the decomposition-width. We provide an explicit family of hypergraphs of large decomposition-width and we prove that every MSO property can be checked in linear time for hypergraphs with bounded decomposition-width when their decomposition is given. Finally, the decomposition-width of a graph is proved to be bounded by twice its clique-width, which suggests that decomposition-width is a generalization of clique-width to relations of large or unbounded arity.

1 Introduction

Hypergraphs are a straightforward generalization of graphs: their edges or *hyperedges* are sets of any number of vertices instead of two. A hypergraph is said to be k-uniform when all its edges are of size k, a 2-uniform hypergraph is thus a graph. Most problems on graphs generalize to hypergraphs and often become harder to solve. For instance, a matching or a spanning tree in a graph can be computed in polynomial time while computing an exact cover in a 3-uniform hypergraph or a spanning hypertree in a 4-uniform hypergraph is NP-hard [1].

Numerous combinatorial structures are nothing but hypergraphs satisfying some properties, like the matroids, the convex geometries or the intersecting set families. The matroid intersection problem, which generalizes many problems in combinatorial optimization, is also NP-hard as soon as three matroids are involved.

Besides complexity issues, we have to deal with the representation of hypergraphs. Indeed an explicit representation of a hypergraph over n vertices may be of size exponential in n. With this representation even a linear time algorithm may not be satisfying. A way to address the problem is to consider hypergraphs given by a *membership oracle*, that is a black-box which decides in unit time whether a set is an edge. In practice, we are interested in classes of hypergraphs whose membership oracle is polynomial time decidable, like representable matroids or k-uniform hypergraphs. A tree-representation of small size can also be used when the set of edges of the hypergraph is closed under set operations such as union, intersection,…, see [2].

The notion of treewidth is an important parameter in graph algorithms: it measures how "tree-like" a graph is. Every graph property expressible in monadic second order logic can be decided in linear time for graphs of bounded treewidth [3]. To a relational structure one can associate its Gaifman graph, where a tuple in a relation is replaced by a clique. Some problems are tractable on structures with a Gaifman graph of bounded treewidth but this measure is not appropriate for hypergraphs with edges of unbounded size. Hypertree width and its variants (see [4, 5]) generalize treewidth to hypergraphs. Properties expressible by primitive positive formulas — constraint satisfaction problems — are tractable on hypergraphs of bounded hypertree width, but this does not include all the important problems we have mentioned in the introduction.

The parameters of the previous paragraph are bounded only on sparse graphs or hypergraphs. The clique-width parameter is a good measure of the complexity of dense graphs: a large number of NP-complete problems becomes tractable on instances of bounded clique-width. However, it is not simple to generalize clique-width to uniform hypergraphs as it is explained in [6].

In this article, we want a decomposition which makes MSO properties tractable and which fully represents a hypergraph even when it is dense. The solution we propose to both problems, namely the notion of *decomposition-width*, is partly inspired by successful matroid decompositions [7, 8]. In fact, Král has defined the decomposition-width of matroids in [9], generalizing the notion of branch-width, and it happens to be a specialization of the decomposition we introduce. For space reasons, we cannot present in this article the restriction of decomposition-width to hereditary hypergraphs, matroids and directed matroids, but we refer the interested reader to [10].

The main contributions of the paper are:

- The study of decomposition-width for hypergraphs and its properties.
- The design of an explicit family of exponential decomposition-width thanks to the notion of type of an edge (Theorem 1).
- A linear time model-checking algorithm for MSO on hypergraphs of fixed decomposition-width (Theorem 2).
- The proof that decomposition-width is equivalent to clique-width on graphs thanks to an alternate decomposition for uniform hypergraphs (Theorem 3).

Organization of the paper. In Sect. 2, we introduce a tree-decomposition which represents a hypergraph, and then study the associated decomposition-width parameter in Sect. 3. In particular we bound the maximal decomposition-width of a hypergraph with n vertices and exhibit an explicit hypergraph whose width is close to this upper-bound. The MSO logic for hypergraphs is presented in Sect. 4, and we establish that the formulas of this logic can be checked in linear time over hypergraphs represented by their decomposition. In Sect. 5, we study the restriction of decomposition-width to uniform hypergraphs and show its relationship with clique-width. Finally in Sect. 6, we consider the problem of finding a decomposition of a hypergraph, and we prove that it is NP-hard even in very restricted cases.

Most proofs can be found in the Appendix.

2 Tree Representation of a Hypergraph

In this section, we introduce the representation of edge-labeled hypergraphs by terms. We could represent objects with several second order relations or one relation with arity larger than one, but we stick to hypergraphs to keep the presentation simple.

A functional signature is a pair (F, A), where F is a finite set of function symbols of positive arity and A is a finite set of constants. We denote by $T(F, A)$ the set of terms built over (F, A). Note that a term can be seen as a ranked tree of bounded degree: each internal node is labeled by an element of F, each leaf by an element of A. In this article, we use indifferently both terminologies, for instance the occurrences of the constants in a term are generally called the leaves of the term.

Definition 1. *Let F_t be the set of unary and binary functions of domain and codomain $\{0, \ldots, t\}$. We denote by $\mathcal{H}_t$ the set of terms $T(F_t, \{0, \ldots, t\})$.*

We now explain how a term of $\mathcal{H}_t$ represents a hypergraph, whose edges are labeled by integers (or colored), by means of a suitable substitution of the labels of the leaves.

Definition 2. *Let T be a term of $\mathcal{H}_t$, let L be the set of leaves of T and let X be one of its subsets. The value of the term T where the labels of the leaves not in X are replaced by 0 is called the value of X in T and is denoted by $v(X, T)$.*

The function v associates to each subset of the leaves a value in $\{0, \ldots, t\}$, it can thus be seen as a $(t + 1)$-coloring of the complete hypergraph with $|L|$ vertices. We want the term to represent an unlabeled hypergraph, therefore a set is chosen to be an edge or a non edge according to its color.

Definition 3. *Let T be a term of $\mathcal{H}_t$, let L be the set of leaves of T and let $S \subseteq \{0, \ldots, t\}$. The hypergraph H represented by (T, S) has L for vertices and its set of hyperedges is $\{X \subseteq L \mid v(X, T) \in S\}$.*

Example 1. See Fig. 1 for a hypergraph and its representation, where the functions labeling the nodes are defined as follows:

f_1	0	1
0	1	0
1	0	1

f_2	0	1
0	0	1
1	0	0

f_3	0	1
0	0	0
1	1	0

f_4	0	1
0	0	0
1	0	1

We now give several restrictions of $\mathcal{H}_t$, which do not decrease the number of representable hypergraphs. From these restrictions, we derive a normal form for terms representing hypergraphs, which is easier to manipulate. The set of integers $\{1, \ldots, t\}$ is denoted by $[t]$.

Proposition 1. *Let H be a hypergraph represented by (T, S) with $T \in \mathcal{H}_t$, then H is also represented by $(\tilde{T}, [t])$ with $\tilde{T} \in \mathcal{H}_t$.*

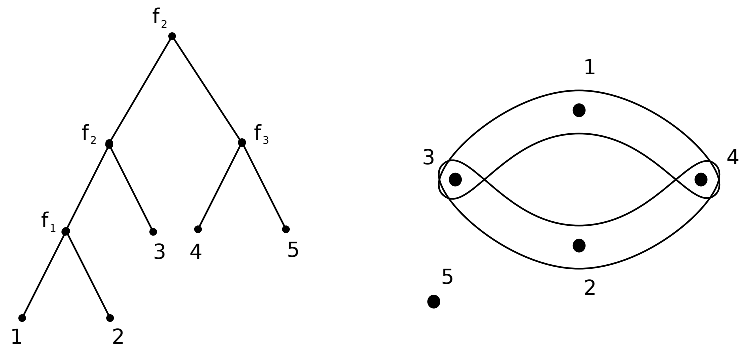

Fig. 1. A term of $\mathcal{H}_1$ and its associated hypergraph

This proposition allows to forget from now on the set S when defining a hypergraph by a term: we assume that, unless it is stated otherwise, $S = [t]$. However, a modification of the set of accepting colors can be useful to prove simple properties. For instance, let H be a hypergraph and $\overline{H}$ the hypergraph over the same vertices such that a set is an edge in $\overline{H}$ if and only if it is not an edge in H. If H is represented by (T, S), then $\overline{H}$ is represented by $(T, \{0, \ldots, t\} \setminus S)$.

It is possible to modify a term of $\mathcal{H}_t$ by simple local transformations of the the functions it contains so that it does not contain unary functions and all its leaves are labeled by 1.

Proposition 2. *A term of $\mathcal{H}_t$ is in normal form if it contains only binary functions and it has only* 1 *as constants. Let H be a hypergraph with two or more vertices represented by (T, S) where $T \in \mathcal{H}_t$, then there is $\tilde{T} \in \mathcal{H}_t$ in normal form such that $(\tilde{T}, [t])$ represents H.*

3 Properties of the Decomposition-width

The representation of a hypergraph by a term in $\mathcal{H}_t$ can be seen as a tree decomposition of this hypergraph. We say that a term in $\mathcal{H}_t$ is of *width* t. A very similar notion of width but restricted to matroids has been introduced in [9] under the name decomposition-width, therefore we use the same name.

Definition 4. *Let H be a hypergraph. The decomposition-width of H, denoted by* $\mathrm{dw}(H)$*, is the smallest integer t such that H is represented by a term of $\mathcal{H}_t$.*

3.1 Operations on Hypergraphs and Decomposition-width

In this subsection, we study the behavior of the decomposition-width with regards to several operations on hypergraphs. One of the most simple is the restriction of a hypergraph to a subset of its vertices, which admits several variants defined below.

Definition 5. *Let* $H = (V, E)$ *be a hypergraph, and let* $W \subseteq V$*. The* section hypergraph $H \times W$ *is defined as* $(W, \{e \in E \mid e \subseteq W\})$*. The* subhypergraph H_W *is defined as* $(W, \{e \cap W \mid e \in E\})$*.*

The value of some graph parameters, such as the tree-width or the clique-width, does not increase when considering subgraphs. We prove the same property for the decomposition-width and the section hypergraph.

Proposition 3. *Let* $H = (V, E)$ *be a hypergraph and* $W \subseteq V$ *then* $\mathrm{dw}(H \times W) \leq \mathrm{dw}(H)$*.*

For a subhypergraph, a similar result does not seem to hold, unless the set of vertices which induces the subhypergraph is chosen according to the decomposition of the hypergraph.

Proposition 4. *Let* T *be a term of* $\mathcal{H}_t$ *which represents a hypergraph* H*. Let* W *be the set of leaves of some subtree* T' *of* T*. There is a set* $S \subseteq \{0, \ldots, t\}$ *such that* (T', S) *represents the hypergraph* H_W*.*

We have seen that the removal of a vertex leaves the decomposition-width unchanged. In the next proposition, we prove that the removal or addition of an edge increase the decomposition-width of at most 1.

Proposition 5. *Let* H *be a hypergraph and let* e_1 *be an edge of* H *then* $\mathrm{dw}(H \setminus \{e_1\}) \leq \mathrm{dw}(H) + 1$*. If* e_1 *is not an edge then* $\mathrm{dw}(H \cup \{e_1\}) \leq \mathrm{dw}(H) + 1$*.*

Remark 1. The inequality in the proposition is tight, otherwise all hypergraphs would be of decomposition-width 1, a fact which is proved false in the next subsection.

Proposition 6. *Let* H_1 *and* H_2 *be two hypergraphs and* $H_1 \cup H_2$ *their disjoint union (or direct sum) and let* $m = \max(\mathrm{dw}(H_1), \mathrm{dw}(H_2))$*. The following inequalities hold:* $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m + 1$*.*

Remark 2. Any hypergraph with two disjoint edges has decomposition-width 2, while a hypergraph with one edge has decomposition-width 1. The inequalities of the last proposition are thus tight.

We can also bound the decomposition-width of the union of two hypergraphs which share a point a by two plus the maximum of their decomposition-width. Substitute the term representing one hypergraph to the constant representing a in the second term. Two additional colors are used to denote the empty set and $\{a\}$. This does not work when the hypergraphs share more than one point.

3.2 Bounds on the Decomposition-width

In this subsection, we try to determine what is the largest decomposition-width of a hypergraph with n vertices.

Lemma 1. *Let H be a hypergraph with n vertices and k hyperedges denoted by $e_1, \dots, e_k$, there is a term T of $\mathcal{H}_k$ whose leaves are in bijection with the vertices of H such that $v(e_i, T) = i$ and if X is not an edge, $v(X, T) = 0$.*

From the previous lemma, we derive a bound on the decomposition-width of a hypergraph with n vertices.

Proposition 7. *Let H be a hypergraph with n vertices, then* $\mathrm{dw}(H) \leq 2^{\lceil \frac{n}{2} \rceil}$.

As a corollary of Proposition 7, all finite hypergraphs have a finite decomposition-width. We now prove that the decomposition-width is unbounded, by studying $H_{k,n}$ the hypergraph with n vertices and all edges of size k. Let T be a term whose functions satisfy $f(x, y) = x+y$ when $x+y \leq k$ and $f(x, y) = k+1$ otherwise. The hypergraph $H_{k,n}$ is represented by $(T, \{k\})$ and thus $\mathrm{dw}(H_{k,n}) \leq k+1$. To prove a lower bound on the decomposition-width of $H_{k,n}$, we define the notion of type of an edge and we show that a hypergraph with a lot of different types cannot be easily decomposed.

Definition 6. *Let $H = (V, E)$ be a hypergraph and let $X, Y \subseteq V$ with $X \cap Y = \emptyset$. The type of X with regard to Y, denoted by* $\mathrm{type}(X, Y)$ *is the set* $\{W \subseteq Y \mid X \cup W \in E\}$. *Let $\overline{X}$ be the complement of X, we let* $\mathrm{Type}(X)$ *be the set* $\{\mathrm{type}(Z, \overline{X}) \mid Z \subseteq X\}$.

Lemma 2. *Let $T \in \mathcal{H}_t$ be a term which represents the hypergraph H and let T' be one of its subterm. Let L be the set of vertices in T' then* $|\mathrm{Type}(L)| \leq t + 1$.

Proof. Let $X, Y \subseteq L$, such that $v(X, T') = v(Y, T')$. The type of X and Y are the same with regard to $\overline{L}$ by definition of a representation by a term. Hence there must be at least as many values for $v(X, T')$, for all $X \subseteq L$, as elements in $\mathrm{Type}(L)$. ◻

Thanks to the previous lemma, we can prove that for all $n > 3k$, we have $\mathrm{dw}(H_{k,n}) = k + 1$. In fact we can compute the decomposition-width of $H_{k,n}$ exactly for all k and n: up to a small additive constant, we have $\mathrm{dw}(H_{k,n}) = \frac{n}{3}$ for $\frac{3}{2}k \leq n \leq 3k$ and $\mathrm{dw}(H_{k,n}) = n - k$ for $k \leq n \leq \frac{3}{2}k$. We have proved that the hypergraph $H_{n,3(n+1)}$ with $3(n+1)$ vertices has a decomposition-width $n + 1$. Therefore hypergraphs represented by terms of $\mathcal{H}_t$ are strictly contained in the hypergraphs represented by terms of $\mathcal{H}_{t+1}$. We would like to improve this result and prove that there is a hypergraph whose decomposition-width is superpolynomial in the number of its vertices.

Proposition 8. *For $n \geq 8$, there is a hypergraph H with n vertices such that* $\mathrm{dw}(H) > \frac{2^{\lceil \frac{n}{2} \rceil}}{n}$.

The proof is simply by counting the terms in normal form of $\mathcal{H}_t$. The result of Proposition 8 is not entirely satisfying, since the hypergraph which has a superpolynomial decomposition-width is not given explicitly, i.e., with a polynomial time membership oracle. Our aim is to find a family of hypergraphs with a large width, which could play the same role the grids do for treewidth and clique-width.

Definition 7. *Let I_n be the hypergraph with set of vertices $[n]$ and such that $X \subseteq [n]$ is an edge if and only if $|X| \in X$.*

A set is an edge in I_n if it contains the vertex indexed by the cardinal of the set. Remark that one can decide if a set is an edge in linear time. This hypergraph has an exponential number of edges and is thus a good candidate for having a large decomposition-width: we prove that it is exponential in $O(n)$.

Theorem 1. *For all $n > 0$, we have* $\mathrm{dw}(I_n) \geq 2^{\frac{n}{27}}$.

Proof. Let T be a term which represents I_n. There is a subterm T' of T, such that L, the set of leaves of T', satisfies the inequalities: $n/3 \leq |L| \leq 2n/3$. One of the intersection of L with $\{1, \dots, n/3\}$, $\{n/3+1, \dots, 2n/3\}$ and $\{2n/3+1, \dots, n\}$ is of cardinal larger than $n/9$, we assume it is $|L \cap \{n/3+1, \dots, 2n/3\}| \geq \frac{n}{9}$, the two other cases are similar and a bit simpler.

Let $S = L \cap \{\frac{n}{3}+1, \dots, \frac{2n}{3} - \frac{n}{27}\}$, we have $|S| \geq 2n/27$. W.l.o.g. assume that $|S| = 2n/27$, otherwise replace S by any of its subsets of this cardinal. We denote by C a set of elements of cardinal $n/3 - 2n/27$ in $L \setminus S$. Let us consider two distinct sets X and Y of elements in S such that $|X| = |Y| = n/27$. W.l.o.g. there is an element $i \in S$ such that $i \in X$ but $i \notin Y$. Remark that $X \cup C$ and $Y \cup C$ are of cardinality $n/3 - n/27$ and that $0 \leq i - (n/3 - n/27) \leq n/3$. Hence there is a subset Z of $\overline{L}$ such that $|Z| = i - (n/3 - n/27)$. Therefore $X \cup C \cup Z$ is an edge of I_n while $Y \cup C \cup Z$ is not, which implies that $\mathrm{type}(X \cup C, L) \neq \mathrm{type}(Y \cup C, L)$. We have $\binom{2n/27}{n/27}$ sets of size $n/27$ in S thus $|\operatorname{Type}(L)| \geq \binom{2n/27}{n/27} \geq 2^{\frac{n}{27}}$. By Lemma 2, the previous inequality completes the proof. $\square$

4 Model-checking of MSO on Hypergraphs of Fixed Decomposition-width

4.1 The Logic MSO for Hypergraphs

A MSO formula for a hypergraph is built from the classical logic operators (the negation, the disjunction and the conjunction) and the quantifications over first and second order variables which denote respectively vertices and sets of vertices of the hypergraph. It also contains the equality predicate, the predicate of membership of an element in a set and the edge predicate $E(X)$ which is true when X is an edge of the hypergraph.

We now give several examples of hypergraphs properties that can be expressed in the MSO logic. The formula $\forall X E(X)$ defines the complete hypergraph. We can state that a hypergraph is a clutter, that is no edge is contained in another edge: $\forall X, Y[(X \subset Y \wedge E(Y)) \Rightarrow \neg E(X)]$. The two other axioms which characterize the set of circuits of a matroid are expressible in MSO, thus there is a formula which holds if and only if a hypergraph is the set of circuits of a matroid.

A k-coloring of the hypergraph $H = (V, E)$ is a function f from V to $[k]$ such that for each edge $e \in E$ there are $v_1, v_2 \in e$ such that $f(v_1) \neq f(v_2)$. The property to be k-colorable can be expressed by the following MSO formula:

$$\exists X_1 \ldots \exists X_k \bigwedge_{i \neq j}(X_i \cap X_j = \emptyset) \wedge \forall X E(X) \Rightarrow$$
$$[\exists v_1 \exists v_2 (v_1 \in X) \wedge (v_2 \in X) \wedge \bigvee_{i \neq j}(v_1 \in X_i) \wedge (v_2 \in X_j)]$$

Interesting objects in hypergraphs can be captured by a MSO formula with a free variable. A hitting set or transversal is a set of vertices which meets all edges of a hypergraph. A set X is a transversal if and only if the following formula holds $\textsc{Transversal}(X) \equiv \forall Y[E(Y) \Rightarrow (X \cap Y \neq \emptyset)]$. Usually, we are interested in minimal transversals, captured by the formula $\textsc{Transversal}(X) \wedge \forall Y[\textsc{Transversal}(Y) \Rightarrow \neg(Y \subsetneq X)]$.

4.2 Representation of the Hyperedge Relation

In this subsection, we want to obtain an algorithm that checks whether a MSO formula ϕ holds on a hypergraph given by a term $T \in \mathcal{H}_t$. To do so we explain how to transform ϕ into a MSO formula on T. This proof can be seen as the construction of a tree automata which checks the formula ϕ by running on T.

We first need to recall how a term of $T(F, A)$ is represented by a relational structure. The domain of the structure is the set of nodes of the term. The structure has the binary relation $lchild(x, y)$ (resp. $rchild(x, y)$) which is true when y is the left child of x (resp. the right child of x). There is also one unary relation for each symbol in F and A, denoted by $label(s) = e$, which holds when e is the label of the node s.

Let T be a term in normal form of $\mathcal{H}_t$ which represents a hypergraph H. We prove how to represent the atomic formula $E(X)$ by a formula on T. For each node s, let us denote by T_s the subterm rooted at s and L_s its set of leaves. For each subterm T_s, we want to compute the integer $v(X \cap L_s, T_s)$. Remark that if we are able to compute $v(X, T)$ it is easy to decide whether X is an edge of H.

To represent the values of X in the different subterms, let us introduce the second-order variables C_i for $i = 0, \ldots, t$ and let denote their set by $\boldsymbol{C}$. We now design in several steps a formula such that $C_i(s)$ holds if and only if $v(X \cap L_s, T_s) = i$.

The formula $\Omega(\boldsymbol{C}) \equiv \forall s \bigvee_{i=0}^{t} \left(C_i(s) \wedge \bigwedge_{j \neq i} \neg C_j(s) \right)$ states that $\boldsymbol{C}$ represents one and only one value of X in T_s for each node s.

Now remark that the value of a set in a term is defined locally. For each node s with label f, children s_1 and $_2$, we have the equality $v(X \cap L_s, T_s) = f(v(X \cap L_{s_1}, T_{s_1}), v(X \cap L_{s_2}, T_{s_2}))$. We can enforce these equalities by the following formula:

$$\begin{aligned} &\Psi_1(\boldsymbol{C}) \equiv \forall s \neg leaf(s) \Rightarrow [\exists s_1, s_2\, lchild(s, s_1) \wedge rchild(s, s_2) \wedge \\ &\bigwedge_{f,i,j,k} ((label(s) = f \wedge C_i(s_1) \wedge C_j(s_2) \wedge C_k(s)) \Rightarrow f(i,j) = k)] \end{aligned}$$

The previous formula "computes" the value of X at internal nodes of T. At the leaves, the value of X is 1 if the leaf is in X and 0 otherwise, a fact we express by $\Psi_2(\boldsymbol{C}, X) \equiv \forall s\, leaf(s) \Rightarrow [(s \in X \Rightarrow C_1(s)) \wedge (s \notin X \Rightarrow C_0(s))]$.

By definition, X is a hyperedge of H represented by T if $v(X,T) \in [t]$, which is expressed by $\Psi_3(\boldsymbol{C}) \equiv \exists r\, root(r) \wedge \bigvee_{i\in[t]} C_i(r)$.

Let G be the function that associates to the formula ϕ in MSO over hypergraphs the formula $G(\phi)$ by relativization to the leaves and replacement of the predicate $E(X)$ by the formula $\exists C_0 \ldots \exists C_t\, [\Psi_1(\boldsymbol{C}) \wedge \Psi_2(\boldsymbol{C}, X) \wedge \Psi_3(\boldsymbol{C})]$. Let H be a hypergraph represented by the term T and let ϕ be a MSO formula, by construction $(H, \boldsymbol{a}) \models \phi(\boldsymbol{x}) \Leftrightarrow (T, \boldsymbol{a}) \models G(\phi(\boldsymbol{x}))$. Since the model-checking of MSO property can be done in linear time on terms [11], we obtain the following theorem.

Theorem 2. *Let ϕ be a* MSO *formula of size l and H a hypergraph with n vertices given by a term of $\mathcal{H}_t$. There is an algorithm which decides whether $H \models \phi$ in time $f(t,l) \times n$ where f is a computable function.*

To decide whether a hypergraph is k-colorable is NP-complete for $k \geq 3$. We have given a MSO formula which holds if and only if a hypergraph is k-colorable, thus this problem is linear time decidable over $\mathcal{H}_t$. We can also decide whether a hypergraph given by a term of $\mathcal{H}_t$ is the set of circuits of a matroid in linear time. However a direct method is given in [9] with a linear time complexity and a small constant.

Finally, it is hard to list all minimal transversals of a hypergraph, a problem with important applications to database, boolean circuits and I.A. [12]. Using Proposition 1 of [13] and our reduction, we obtain an algorithm which outputs all minimal transversals, in a time linearly dependent on their number, for hypergraphs given by a term of $\mathcal{H}_t$.

5 Uniform Hypergraphs

In this section, we prove that the decomposition-width parameter is already well-known on a restricted class of hypergraphs: it is equivalent up to a factor 2 to the clique-width on graphs. Let us recall the definition of *clique-width*. Let $\mathcal{L}$ be a set of labels, a labeled graph is a pair (G, γ) where γ is a function from V to $\mathcal{L}$. Let $F_{\mathcal{L}}$ be the following set of graph operations:

- The disjoint union of two labeled graphs is denoted by $\oplus$.
- For all $a, b \in \mathcal{L}$, let $\rho_{a\to b}$ be the function which renames every vertex labeled by a into b.
- For all $a, b \in \mathcal{L}$, let $\eta_{a,b}$ be the function which adds all edges between the vertices labeled a and those labeled b.

For all $a \in \mathcal{L}$, let G_a be the graph with one vertex labeled by a and $G_{\mathcal{L}} = \{G_a \mid a \in \mathcal{L}\}$.

Definition 8. *The clique-width of the graph G, denoted by* $\mathrm{cw}(G)$*, is the minimum of the $n \in \mathbb{N}$ such that $\exists \gamma, (G, \gamma) \in T(F_{[n]}, G_{[n]})$. A term of $T(F_{[n]}, C_{[n]})$ is called an n-expression.*

We now introduce an alternate decomposition on uniform hypergraph and prove it is related to clique-width.

Definition 9. *Let k and t be two integers and let $D = \{(0,0),(k,0),(k+1,0)\} \cup \{(i,j)\}_{0<i<k,0\leq j\leq t}$. Let $\mathcal{F}_{k,t}$ be the set of unary and binary functions with domain and codomain D which satisfy for all $(a,b),(c,d) \in D^2$:*

- *$f((a,b)) = (a,c)$ for some $c \leq t$*
- *$g((a,b),(c,d)) = (a+c,e)$ for some $e \leq t$ when $a+c<k$*
- *$g((a,b),(c,d)) = (k,0)$ or $(k+1,0)$ when $a+c=k$*
- *$g((a,b),(c,d)) = (k+1,0)$ when $a+c>k$*

Definition 10. *We denote by $\mathcal{H}_{k,t}$ the set of terms $T(\mathcal{F}_{k,t},\{(1,i)\}_{0\leq i\leq t})$. They represent hypergraphs as in Definition 3. The uniform decomposition-width of a k-uniform hypergraph H, denoted by $\mathrm{dw}_u(H)$, is the smallest t such that H is represented by a term of $\mathcal{H}_{k,t}$.*

The definition is such that, if an edge has value (i,j), its cardinal is i. The only exception is $(k+1,0)$ which denotes any set of cardinality larger or equal to k which is not an edge.

Proposition 9. *All hypergraphs represented by a term of $\mathcal{H}_{k,t}$ are k-uniform and for any k-uniform graph H, the following holds:*

$$\mathrm{dw}_u(H) \leq \mathrm{dw}(H) \leq (k-1)(\mathrm{dw}_u(H)+1)+2.$$

The proof of Proposition 9 relies on a easy to compute normalization of a term of $\mathcal{H}_t$ representing a k-uniform hypergraph into a term of $\mathcal{H}_{k,t}$.

Proposition 10. *Let G be a graph, then $\mathrm{dw}_u(G) \leq \mathrm{cw}(G) \leq 2\,\mathrm{dw}_u(G)$.*

Proof. ($\mathrm{dw}_u(G) \leq \mathrm{cw}(G)$). Let T be a t-expression which represents a vertex-colored graph G, we inductively build a term $\tilde{T}$ of $\mathcal{H}_{2,t}$ such that $(\tilde{T},\{1\})$ also represents G. Our induction hypothesis states that for any vertex u of color i in G, $v(\{u\},\tilde{T}) = (1,i)$ and for any $\{u,w\}$ edge of G, $v(\{u,w\},\tilde{T}) = (2,0)$.

Let T be the constant G_i, it represents a graph with one vertex of color i. The constant $(1,i)$ satisfies the induction hypothesis and is in $\mathcal{H}_{2,t}$.

Let $T = T_1 \oplus T_2$, the induction hypothesis gives two terms $\tilde{T_1}$ and $\tilde{T_2}$ of $\mathcal{H}_{2,t}$ which represent the same vertex-colored graphs as T_1 and T_2. Let $f \in \mathcal{F}_{2,t}$, such that $f((1,i),(1,j)) = (3,0)$ and $f((1,i),(0,0)) = f((0,0),(1,i)) = (1,i)$. The term $\tilde{T} = f(T_1,T_2)$ is in $\mathcal{H}_{2,t}$ and represents $T = T_1 \oplus T_2$.

Let $T = \rho_{i\to j}T_1$ and let $\tilde{T_1}$ bet the term of $\mathcal{H}_{2,t}$ which, by induction hypothesis, represents the same vertex-colored graph as T_1. Let $f((1,i)) = (1,j)$, f is the identity otherwise. We let $\tilde{T} = f(T_1)$, it is a term of $\mathcal{H}_{2,t}$ and it represents the same graph as $T = \rho_{i\to j}T_1$.

Let $T = \eta_{i,j}T_1$ and let $\tilde{T_1}$ be the term of $\mathcal{H}_{2,t}$ which, by induction hypothesis, represents the same vertex-colored graph as T_1. For each pair of leaves (u,w) of $\tilde{T_1}$ such that $v(\{u\},\tilde{T_1}) = (1,i)$ and $v(\{w\},\tilde{T_1}) = (1,j)$ there is exactly one subterm of $\tilde{T}_1$, which can be written $f(\tilde{T}_2, \tilde{T}_3)$ and u is a leaf of $\tilde{T}_2$ and w is a leaf of $\tilde{T}_3$. Let $\tilde{f}(v(\{u\}, \tilde{T}_2), v(\{w\}, \tilde{T}_3)) = (2,0)$ and $\tilde{f}(x,y) = f(x,y)$ otherwise. We let $\tilde{T}$ be the term $\tilde{T}_1$ where each function f which corresponds to a pair (u,w) is replaced by $\tilde{f}$. By construction $\tilde{T} \in \mathcal{H}_{2,t}$ and it represents the same graph as T.

($\mathrm{cw}(G) \leq 2\,\mathrm{dw}_u(G)$). Let T be a term of $\mathcal{H}_{2,t}$, we prove by induction on T that there is a $2t$-expression which represents the same colored graph as T. Assume that T is in normal form, then either T is the constant $(1,i)$ and the $2t$-expression G_i represents the same colored graph or $T = f(T_1, T_2)$. By induction, we get two $2t$-expressions, e_1 and e_2 which represents the same colored graphs as T_1 and T_2. We show that we can simulate the action of f through the three operations of t-expressions. We use twice more colors to differentiate elements in e_1 and e_2 which represent t-colored graphs by the induction hypothesis. Let ρ_l and ρ_r be two compositions of relabelings, such that a vertex of color i in e_1 is relabeled into (l,i) and a vertex of color i in e_2 is relabeled into (r,i). We then consider the $2t$-expression $e = (\rho_l e_1) \oplus (\rho_r e_2)$.

For each pair of colors (i,j), $f((1,i),(1,j))$ is either equal to $(2,0)$ or $(3,0)$, the first denotes an edge between the vertices of color i and j. Let η be the composition of operators which adds all edges between vertices of colors (l,i) and (r,j) such that $f(1,i),(1,j)) = (2,0)$.

For each $i \leq t$, we have $f((1,i),(0,0)) = (1,j)$, that is f changes the vertices of color i into vertices of color j. We can simulate that by applying the relabeling of the color (l,i) into j. Let ρ_1 be the composition of relabelings such that each color (l,i) is changed into j, as explained previously. Let ρ_2 be the composition of relabelings changing the colors (r,i) into j where j is given by $f((0,0),(1,i))$.

The reader may now easily check that the $2t$-expression $\rho_1 \circ \rho_2 \circ \eta(e)$ represents the same labeled graph as T. $\square$

Theorem 3. *Let G be a graph, then* $\mathrm{cw}(G)/2 \leq \mathrm{dw}(G) \leq \mathrm{cw}(G) + 2$.

6 Computing the Decomposition of a Hypergraph

In this article, we have always considered hypergraphs given by a term of $\mathcal{H}_t$. We prove that, if the hypergraph is given in some other way, computing its decomposition-width, and thus a good representation, is hard. First, let us assume that we want to compute the decomposition-width of k-uniform hypergraphs represented by the list of their at most n^k edges. By Theorem 3, the decomposition-width is equivalent to clique-width on graphs, which is NP-hard to approximate [14], we thus obtain the following hardness result.

Theorem 4. *Let k be an integer, the problem of computing the decomposition-width of a k-uniform hypergraphs is* NP*-hard.*

Now assume that the hypergraph is given by a membership oracle, that is we can test whether a set is an edge or not. From several queries, we want to learn a good decomposition of the hypergraph. Notice that a term of $\mathcal{H}_1$ is a read-once formula (each variable appears only once) built from all possible logical connectors. Read-once formulas built from the connectors AND, OR and NOT cannot be learned in polynomial time with only membership queries (see [15] and the references therein), which yields the following theorem.

Theorem 5. *There is no polynomial time algorithm to compute the decomposition of a hypergraph of decomposition-width* 1 *given by a membership oracle.*

On the other hand, a read-once formula can be learned in polynomial time with a polynomial number of queries to a membership oracle and an equivalence oracle (see [16]). The k-uniform hypergraphs have a polynomial time membership and equivalence oracle, hence we can find their decomposition in polynomial time when they are of decomposition-width 1.

References

1. Duris, D., Strozecki, Y.: The complexity of acyclic subhypergraph problems. Workshop on Algorithms and Computation (2011) 45–56
2. Bui-Xuan, B.M., Habib, M., Rao, M.: Tree-representation of set families and applications to combinatorial decompositions. European Journal of Combinatorics (2011)
3. Courcelle, B.: The monadic second-order logic of graphs. III: Tree-decompositions, minors and complexity issues. Informatique théorique et applications **26**(3) (1992) 257–286
4. Gottlob, G., Leone, N., Scarcello, F.: A comparison of structural csp decomposition methods. Artificial Intelligence **124**(2) (2000) 243–282
5. Adler, I., Gottlob, G., Grohe, M.: Hypertree width and related hypergraph invariants. European Journal of Combinatorics **28**(8) (2007) 2167–2181
6. Adler, H., Adler, I.: A note on clique-width and tree-width for structures. Arxiv preprint arXiv:0806.0103 (2008)
7. Hliněný, P.: Branch-width, parse trees, and monadic second-order logic for matroids. J. Comb. Theory, Ser. B **96**(3) (2006) 325–351
8. Strozecki, Y.: Monadic second-order model-checking on decomposable matroids. Discrete Applied Mathematics **159**(10) (2011) 1022–1039
9. Kral', D.: Decomposition Width of Matroids. International Conference on Automata, Languages and Programming (2010) 55–66
10. Strozecki, Y.: Enumeration complexity and matroid decomposition. PhD thesis, Université Paris Diderot - Paris 7 (2010)
11. Thatcher, J., Wright, J.: Generalized finite automata theory with an application to a decision problem of second-order logic. Theory of Computing Systems **2**(1) (1968) 57–81
12. Eiter, T., Gottlob, G.: Hypergraph transversal computation and related problems in logic and ai. Logics in Artificial Intelligence (2002) 549–564
13. Courcelle, B.: Linear delay enumeration and monadic second-order logic. Discrete Applied Mathematics **157**(12) (2009) 2675–2700
14. Fellows, M., Rosamond, F., Rotics, U., Szeider, S.: Clique-width minimization is NP-hard. In: Proceedings of the 38th annual ACM symposium on Theory of computing, ACM (2006) 362
15. Angluin, D., Hellerstein, L., Karpinski, M.: Learning read-once formulas with queries. Journal of the ACM (JACM) **40**(1) (1993) 185–210
16. Bshouty, N., Hancock, T., Hellerstein, L.: Learning boolean read-once formulas over generalized bases. Journal of Computer and System Sciences **50**(3) (1995) 521–542

Appendix

In this Appendix are given all the missing proofs.

Section 2:

Proposition 1. *Let H be a hypergraph represented by (T, S) with $T \in \mathcal{H}_t$, then H is also represented by $(\tilde{T}, [t])$ with $\tilde{T} \in \mathcal{H}_t$.*

Proof. Let $\tilde{T}$ be the term T where the function labeling the root is composed with a function which maps the elements not in S to 0. The hypergraph represented by $(\tilde{T}, [t])$ is the same as the one represented by (T, S). $\square$

Proposition 2. *A term of $\mathcal{H}_t$ is in normal form if it contains only binary functions and it has only 1 as constants. Let H be a hypergraph with two or more vertices represented by (T, S) where $T \in \mathcal{H}_t$, then there is $\tilde{T} \in \mathcal{H}_t$ in normal form such that $(\tilde{T}, [t])$ represents H.*

Proof. Let first prove that we can remove unary functions from a term. Each subterm of T of the form $g(f(T_1), T_2)$ can be replaced by the subterm $h(T_1, T_2)$ where $h(x, y) = g(f(x), y)$. We do the same for $g(T_1, f(T_2))$. If there are two vertices or more, there is at least a function of arity 2 in T. Thus $T = f_1 \circ \ldots f_l(g(T_1, T_2))$ and we replace it by $h(T_1, T_2)$ where $h = f_1 \circ \ldots f_l \circ g$. We now apply the previous transformation top-down, until there are no more unary functions and call the result $\tilde{T}$. Since the transformation preserves the value of any set of leaves, the hypergraphs represented by T and $\tilde{T}$ are the same.

We now prove that all constants can be replaced by 1. Let $g(c, T_1)$ be a subterm of T where c is a constant. We replace it by $h(1, T_1)$ where $h(x, y) = g(f(x), y)$ and f maps c to 1. We do the symmetric transformation for $g(T_1, c)$ and we obtain a term $\tilde{T}$ whose constants are 1 only. $\square$

Section 3:

Proposition 3. *Let $H = (V, E)$ be a hypergraph and $W \subseteq V$ then* $\mathrm{dw}(H \times W) \leq \mathrm{dw}(H)$.

Proof. Let T be a term in normal form representing H and let $a \in V$. Assume that $W = V \setminus \{a\}$: it is enough to prove the proposition in this case, we then conclude by removing points successively.

Some subterm of T is of the form $g(a, T_1)$. Let $h(y) = g(0, y)$ and let $\tilde{T}$ be the tree T where $g(a, T_1)$ is changed by $h(T_1)$. By construction the value of any set which does not contain a is the same in T and $\tilde{T}$. Therefore $\tilde{T}$ represents $H \times W$ and $dw(H \times W) \leq dw(H)$. $\square$

Proposition 5. *Let H be a hypergraph and let e_1 be an edge of H then* $\mathrm{dw}(H \setminus \{e_1\}) \leq \mathrm{dw}(H) + 1$. *If e_1 is not an edge then* $\mathrm{dw}(H \cup \{e_1\}) \leq \mathrm{dw}(H) + 1$.

Proof. Let $t = \mathrm{dw}(H)$ and let T be a term of $\mathcal{H}_t$ in normal form which represents H. Let $\tilde{T}$ be a term of $\mathcal{H}_{t+1}$ defined from T as follows. First label every leaf in e_1 by the integer $t+1$. Then, for every subterm $f(T_1, T_2)$ of T, we define $\tilde{f}(t+1, t+1) = t+1$. Let Y be the intersection of e_1 with the leaves of T_1, we define for all i, $\tilde{f}(t+1, i) = f(v(Y, T_1), i)$ and we define in the same way $\tilde{f}(i, t+1)$. On every other values $\tilde{f} = f$. By a simple structural induction, we have for all $X \subseteq V$ and $X \neq e_1$, $v(X, T) = v(X, \tilde{T})$ while $v(e_1, \tilde{T}) = t+1$. The theorem is proved since $(\tilde{T}, [t])$ represents $H \setminus \{e_1\}$ and $(\tilde{T}, [t+1])$ represents $H \cup \{e_1\}$. ☐

Proposition 6. *Let H_1 and H_2 be two hypergraphs and $H_1 \cup H_2$ their disjoint union (or direct sum) and let $m = \max(\mathrm{dw}(H_1), \mathrm{dw}(H_2))$. The following inequalities hold: $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m+1$.*

Proof. Let T_1 and T_2 be two terms representing H_1 and H_2 and which belong to $\mathcal{H}_m$. First let $\tilde{T}_1$ and $\tilde{T}_2$ be the two terms of $\mathcal{H}_{m+1}$ which represent the same hypergraphs as T_1 and T_2 except that $v(\emptyset, T_1) = v(\emptyset, T_2) = m+1$. We obtain these two hypergraphs by the transformation used in the proof of Proposition 5.

Let f be the function from $\{0, \dots, m+1\}^2$ to $\{0, 1\}$ defined by, for all $i > 0$, $f(i, m+1) = f(m+1, i) = 1$ and f is zero on the other couples. The term $f(\tilde{T}_1, \tilde{T}_2)$ represents $H_1 \cup H_2$ therefore $\mathrm{dw}(H_1 \cup H_2) \leq m+1$.

Let V_1 and V_2 denote the set of vertices of H_1 and H_2 respectively. It holds that $H_1 = (H_1 \cup H_2) \times V_1$ and $H_2 = (H_1 \cup H_2) \times V_2$. By Proposition 3, one obtains that $\mathrm{dw}(H_1 \cup H_2) \geq \mathrm{dw}(H_1)$ and $\mathrm{dw}(H_1 \cup H_2) \geq \mathrm{dw}(H_2)$, which concludes the proof. ☐

Lemma 1. *Let H be a hypergraph with n vertices and k hyperedges denoted by $e_1, \dots, e_k$, there is a term T of $\mathcal{H}_k$ whose leaves are in bijection with the vertices of H such that $v(e_i, T) = i$ and if X is not an edge, $v(X, T) = 0$.*

Proof. Proof by induction on k, the number of edges. The hypergraph with n vertices and no edges is of decomposition-width 0. Assume the property is true for k: we have a term T such that $v(e_i, T) = i$ for all $i \leq k$. By Proposition 5, we build a term $\tilde{T}$ such that $v(e_{k+1}, T) = k+1$ which completes the proof. ☐

Proposition 7. *Let H be a hypergraph with n vertices, then $\mathrm{dw}(H) \leq 2^{\lceil \frac{n}{2} \rceil}$.*

Proof. Let V_1, V_2 be a partition of the vertices of H such that their cardinals are at most $l = \lceil \frac{n}{2} \rceil$. We consider the complete hypergraphs H_1 and H_2 over the vertices V_1 and V_2 respectively. They both have at most 2^l hyperedges. We apply Lemma 1 to find two terms T_1 and T_2 of $\mathcal{H}_l$ which represent H_1 and H_2. Let f be the function defined by $f(v(X_1, T_1), v(X_2, T_2)) = 1$ if $X_1 \cup X_2$ is a hyperedge of H, 0 otherwise. This definition is not ambiguous, since the function $(X_1, X_2) \to (v(X_1, T_1), v(X_2, T_2))$ is injective. The term $f(T_1, T_2) \in \mathcal{H}_l$ represents H, hence $\mathrm{dw}(H) \leq 2^{\lceil \frac{n}{2} \rceil}$. ☐

The following proposition is stated in the text before Proposition 8.

Proposition. *For all* $n > 3k$, *we have* $\mathrm{dw}(H_{k,n}) = k + 1$.

Proof. Let n and k be such that $n > 3k$ and let T be a decomposition of $H_{k,n}$ of width l. We can always find a subterm T' of T, such that L, the set of leaves of T', satisfies the inequalities: $\frac{n}{3} \leq |L| \leq \frac{2n}{3}$. Let X be a set of vertices in L, the type of X with regard to $\overline{L}$ depends only on its cardinal and on the cardinal of $\overline{L}$. Since $\frac{n}{3} > k$, we have $|\,\mathrm{Type}(L)| = k + 2$. By Lemma 2, we have $l \geq k + 1$ for any decomposition of width l thus $\mathrm{dw}(H_{k,n}) \geq k + 1$. The equality is proved by the previously mentioned decomposition. □

The exact computation of $\mathrm{dw}(H_{k,n})$ for $n \leq 3k$ can be done by a precise study of $|\,\mathrm{Type}(L)|$ to prove a lower bound and by using a decomposition in three subterms of the same size to prove an upper bound.

Proposition 8. *For* $n \geq 8$, *there is a hypergraph* H *with* n *vertices such that* $\mathrm{dw}(H) > \frac{2^{\lceil \frac{n}{2} \rceil}}{n}$.

Proof. The proof is by counting the terms in normal formal of $\mathcal{H}_t$. There are exactly $\frac{(2n)!}{n!(n+1)!}$ binary trees. For a given binary tree, there are at most $(t+1)^{n(t+1)^2}$ way to choose the functions at its inner nodes.Hence we have a bound on the number of hypergraphs with n vertices and a decomposition-width less than t. On the other hand, there are 2^{2^n} hypergraphs with n vertices and by a simple computation we derive the proposition. □

Section 5:

Proposition 9. *All hypergraphs represented by a term of* $\mathcal{H}_{k,t}$ *are* k*-uniform and for any* k*-uniform graph* H, *the following holds:*

$$\mathrm{dw}_u(H) \leq \mathrm{dw}(H) \leq (k-1)(\mathrm{dw}_u(H) + 1) + 2.$$

Proof. A simple structural induction proves that a term of $\mathcal{H}_{k,t}$ represents a k-uniform hypergraph.

The inequality $\mathrm{dw}(H) \leq (k-1)(\mathrm{dw}_u(H)+1)+2$ is obvious: Let T be a term of $\mathcal{H}_{k,t}$, the domain D is in bijection with $[0, (k-1)(t+1)+2]$, which enables us to see T as a term of $\mathcal{H}_{k(t+1)+2}$ representing the same hypergraph.

Let T be a term of $\mathcal{H}_t$ in normal form which represents a hypergraph H. We inductively build a term $\tilde{T}$ of $\mathcal{H}_{k,t}$ which satisfies the properties:

1. for all $0 \leq i \leq t$, for all sets of constants X of cardinal $l < k$, $v(X, T) = i \Leftrightarrow v(X, \tilde{T}) = (l, i)$.
2. for all sets X of cardinal k which are edges of H, $v(X, \tilde{T}) = (k, 0)$
3. for all sets X of cardinal larger or equal to k which are not edges of H, $v(X, \tilde{T}) = (k+1, 0)$

Remark that once this is proved we obtain a term $\tilde{T} \in \mathcal{H}_{k,t}$ such that $(\tilde{T}, \{(k, 0)\})$ represents H which yields $\mathrm{dw}_u(H) \leq \mathrm{dw}(H)$.

Assume that T is a leaf, since T is normal form, it is labeled by 1 . The term $\tilde{T}$ is the constant $(1, 1)$.

Assume now that $T = g(T_1, T_2)$, the induction hypothesis gives us two terms $\tilde{T}_1$ and $\tilde{T}_2$ which satisfy the previous properties. Let us define $\tilde{g} \in \mathcal{F}_{k,t}$ which mimics the action of g. Let $a + c < k$, then $\tilde{g}((a, b), (c, d)) = (a + c, g(b + d))$ for all b, d. Let $a + c = k$, for all b, d such that there is an edge X of value $g(b, d)$ in T, $\tilde{g}((a, b), (c, d)) = (k, 0)$. Finally on all other couples (a, b) and (c, d), $\tilde{g}((a, b), (c, d)) = (0, k + 1)$.

It is clear that the term $\tilde{T} = \tilde{g}(\tilde{T}_1, \tilde{T}_2)$ satisfies the induction hypothesis, which completes the proof. $\square$