Myhill-Nerode Methods for Hypergraphs

René van Bevern[1], Michael R. Fellows[2],
Serge Gaspers[3], and Frances A. Rosamond[2]

[1] Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany
`rene.vanbevern@tu-berlin.de`
[2] School of Engineering and IT, Charles Darwin University, Darwin, Australia
{`michael.fellows`,`frances.rosamond`}`@cdu.edu.au`
[3] The University of New South Wales and NICTA, Sydney, Australia,
`sergeg@cse.unsw.edu.au`

Abstract. We introduce a method of applying Myhill-Nerode methods from formal language theory to hypergraphs and show how this method can be used to obtain the following parameterized complexity results.

– Hypergraph Cutwidth (deciding whether a hypergraph on n vertices has cutwidth at most k) is linear-time solvable for constant k.
– For hypergraphs of constant *incidence treewidth* (treewidth of the incidence graph), Hypertree Width and variants cannot be solved by simple finite tree automata. The proof leads us to conjecture that Hypertree Width is W[1]-hard for this parameter.

1 Introduction

This work extends the graph-theoretic analog [7] of the Myhill-Nerode characterization of regular languages to colored graphs and hypergraphs. Thus, we provide a method to derive linear-time algorithms (or to obtain evidence for intractability) for hypergraph problems on instances with bounded *incidence treewidth* (treewidth of the incidence graph). From a parameterized complexity point of view [6], incidence treewidth is an interesting parameter, since it can be bounded from above by canonical hypergraph width measures, like the treewidth of the primal graph [15] and the treewidth of the dual graph [19].

Applying Myhill-Nerode methods to hypergraphs, we obtain various parameterized complexity results, which we summarize in the following. Besides these results for hypergraph problems, our extension of the Myhill-Nerode theorem to colored graphs likely applies to other problems, since colored or annotated graphs allow for more realism in problem modeling and often arise as subproblems when solving pure graph problems. It is also straightforward to use our methods for annotated hypergraphs.

Hypergraph Cutwidth. We first apply our Myhill-Nerode approach to Hypergraph Cutwidth (see Section 3 for a formal definition)—a natural generalization of the NP-complete [10] and fixed-parameter tractable [6] Graph Cutwidth problem, for which several fixed-parameter algorithms are known [1, 3, 8, 9, 20]. Cahoon and Sahni [4] designed algorithms for Hypergraph Cutwidth with $k \leq 2$, with running time $O(n)$ for $k = 1$ and running time $O(n^3)$ for $k = 2$, where n is the number of vertices. For arbitrary k, Miller and Sudborough [17] designed an algorithm with running time $O(n^{k^2+3k+3})$. We suspect that the framework of Nagamochi [18] applies to Hypergraph Cutwidth, giving an $n^{O(k)}$ time algorithm. The algorithm we present here has running time $O(n + m)$ for constant k, thus showing Hypergraph Cutwidth to be *fixed-parameter linear* for the parameter k.

In the context of VLSI design, the Hypergraph Cutwidth problem is known as Board Permutation, and it is related to the gate matrix layout problem and several graph problems; see [17] and references therein. We arrived at the Hypergraph Cutwidth problem when studying the Trellis Width problem, which plays a central role in the maximum likelihood decoding for linear block codes [14, 21]. Kashyap [14] observed that Trellis Width is equivalent to Matroid Pathwidth. The class of matroids with pathwidth at most k is closed under taking matroid minors [14], has bounded branchwidth, and membership in a minor-closed matroid family can be tested in cubic time for matroids with bounded branchwidth [13]. This asserts that, for every constant k, there exists a cubic-time algorithm for Trellis Width. [4] However, the proof that a nonuniform fixed-parameter tractable algorithm (there is a different algorithm for each k) [6] exists is nonconstructive: for each constant k, we only know that an $O(n^3)$-time algorithm exists, but it is unknown what it is, and how its running time depends on k. Using our result for Hypergraph Cutwidth, one can show that Trellis Width is fixed-parameter linear parameterized by the code rank; however, this result is outperformed by a relatively simple linear-time kernelization algorithm.

Hypertree Width. The original Myhill-Nerode theorem can be used both positively and negatively: to show that a language is regular, and to show that a language is not regular. Using our hypergraph Myhill-Nerode analog negatively, we obtain evidence that the problems Hypertree Width, Generalized Hypertree Width, and Fractional Hypertree Width are not *fixed-parameter tractable* with respect to the parameter incidence treewidth t, that is, we conjecture that there are no algorithms for these problems running in time $f(t) \cdot n^c$, where n is the input size, c is a constant, and f is a computable function. It is already known that these problems are unlikely to be fixed-parameter tractable for their standard parameterizations [11, 12, 16]. Our result hints that even if the incidence width is constant, these other width measures cannot be computed efficiently.

Due to space constraints, we defer the proofs to the appendix.

Preliminaries. We use the standard graph-theoretic notions of Diestel [5].

[4] This has also been observed by Navin Kashyap [private communication].

Graph Decompositions. A *tree decomposition* of a graph $G = (V, E)$ is a pair $(\{X_i : i \in I\}, T)$ where $X_i \subseteq V$, $i \in I$, are called *bags* and T is a tree with elements of I as nodes such that:

– for each edge $\{u, v\} \in E$, there is an $i \in I$ such that $\{u, v\} \subseteq X_i$, and
– for each vertex $v \in V$, $T[\{i \in I : v \in X_i\}]$ is a non-empty connected tree.

The *width* of a tree decomposition is $\max_{i \in I} |X_i| - 1$. The *treewidth* of G is the minimum width taken over all tree decompositions of G. The notions of *path decomposition* and *pathwidth* of G are defined the same way, except that T is restricted to be a path.

Hypergraphs. A *hypergraph* H is a pair (V, E), where V is a set of *vertices* and E a multiset of *hyperedges* such that $e \subseteq V$ for each $e \in E$. Let $H = (V, E)$ be a hypergraph. The *primal graph* of H, denoted $\mathcal{G}(H)$, is the graph with vertex set V that has an edge $\{u, v\}$ if there exists a hyperedge in H incident to both u and v. It is sometimes called the Gaifman graph of H. The *incidence graph* of H, denoted $\mathcal{I}(H)$, is the bipartite graph (V', E') with vertex set $V' = V \cup E$ and for $v \in V$ and $e \in E$, there is an edge $\{v, e\} \in E'$ if $v \in e$.

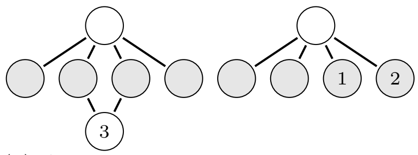
Hypergraph decompositions. Let H be a hypergraph. Generalized hypertree width is defined with respect to tree decompositions of $\mathcal{G}(H)$, however, the width of the tree decompositions is measured differently. Suppose H has no isolated vertices (otherwise, remove them). A *cover* of a bag is a set of hyperedges such that each vertex in the bag is contained in at least one of these hyperedges. The *cover width* of a bag is the minimum number of hyperedges covering it. The *cover width* of a tree decomposition is the maximum cover width of any bag in the decomposition. The *generalized hypertree width* of H is the minimum cover width over all tree decompositions of $\mathcal{G}(H)$.

The *hypertree width* of H is defined in a similar way, except that, additionally, the tree of the decomposition is rooted and a hyperedge e can only be used in the cover of a bag X_i if X_i contains all vertices of e that occur in bags of the subtree rooted at the node i.

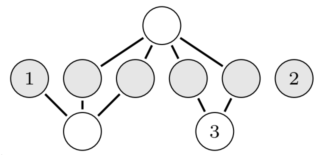
The *fractional hypertree width* of H is defined in a similar way as the generalized hypertree width, except that it uses fractional covers: in a *fractional cover* of a bag, each hyperedge is assigned a non-negative weight, and for each vertex in the bag, the sum of the weights of the hyperedges incident to it is at least 1. The *fractional cover width* of the bag is the minimum total sum of all hyperedges of a fractional cover.

2 Myhill-Nerode for Colored Graphs and Hypergraphs

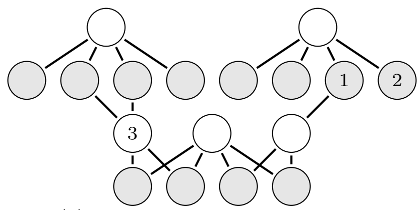
The aim of this section is first to generalize the Myhill-Nerode analog for graphs [7] to colored graphs. From this, we obtain a Myhill-Nerode analog for hypergraphs, since every hypergraph can be represented as its incidence graph with two vertex types: those representing hyperedges and those representing hypergraph-vertices.



(a) A 2-colored 3-boundaried graph G.



(b) A 2-colored 3-boundaried graph H.



(c) The glued graph $G \oplus_c H$.

Fig. 1. Two color-compatible 3-boundaried graphs G and H and their glued graph, where the boundary vertices are marked by their label.

In the last part of the section, we finally describe how our Myhill-Nerode analog yields linear-time algorithms for hypergraph problems. We follow and adapt the notation used by Downey and Fellows [6, Section 6.4].

2.1 Colored Graphs

We now develop an analog of the Myhill-Nerode theorem for colored graphs. The original Myhill-Nerode theorem is stated for languages in terms of concatenations of words. Hence, we clarify what concatenating colored graphs means.

Definition 1. A *t-boundaried graph* G is a graph with t distinguished vertices that are labeled from 1 to t. These labeled vertices are called *boundary vertices*. The *boundary*, $\partial(G)$, denotes the set of boundary vertices of G.

Let G_1 and G_2 be t-boundaried graphs whose vertices are colored with colors from $\{1, \ldots, c_{\max}\}$. We say that G_1 and G_2 are *color-compatible* if the vertices with the same labels in $\partial(G_1)$ and $\partial(G_2)$ have the same color.

For two color-compatible t-boundaried graphs, we denote by $G_1 \oplus_c G_2$ the colored graph obtained by taking the disjoint union of G_1 and G_2 and identifying each vertex of $\partial(G_1)$ with the vertex of $\partial(G_2)$ with the same label, wherein vertex colors are inherited from G_1 and G_2.

Let $\mathcal{U}^{\text{large}}_{t,c_{\max}}$ be the universe of $\{1, \ldots, c_{\max}\}$-colored t-boundaried graphs and $F \subseteq \mathcal{U}^{\text{large}}_{t,c_{\max}}$. We define the *canonical right congruence* $\sim_F$ for F as follows: for $G_1, G_2 \in \mathcal{U}^{\text{large}}_{t,c_{\max}}$, $G_1 \sim_F G_2$ if and only if G_1 and G_2 are color-compatible and for all color-compatible $H \in \mathcal{U}^{\text{large}}_{t,c_{\max}}$, $G_1 \oplus_c H \in F \iff G_2 \oplus_c H \in F$.

The *index* of $\sim_F$ is its number of equivalence classes.

Definition 1 is illustrated in Figure 1. Before we can state our analog of the Myhill-Nerode theorem for colored graphs, we show that every $\{1, \ldots, c_{\max}\}$-colored graph of treewidth at most t can be generated using a constant number of graph operations. To this end, we use the following set of operators. For generating graphs of only one color, the given operators coincide with those given by Downey and Fellows [6, Section 6.4].

Definition 2. The *size-$(t+1)$ parsing operators* for $\{1, \ldots, c_{\max}\}$-colored graphs are:

i) $\{\emptyset_{n_1,\ldots,n_{c_{\max}}} : \sum_{i=1}^{c_{\max}} n_i = t+1\}$ is a family of nullary operators that creates boundary vertices $1, \ldots, t+1$, of which the first n_1 vertices get color 1, the next n_2 vertices get color 2, and so on.
ii) γ is a unary operator that cyclically shifts the boundary. That is, γ moves label j to the vertex with label $j+1 \pmod{t+1}$.
iii) i is a unary operator that assigns the label 1 to the vertex currently labeled 2 and label 2 to the vertex with label 1.
iv) e is a unary operator that adds an edge between the vertices labeled 1 and 2.
v) $\{u_\ell : 1 \leq \ell \leq c_{\max}\}$ is a family of unary operators that add a new vertex of color ℓ and label it 1, unlabeling the vertex previously labeled 1.
vi) $\oplus_{\mathrm{c}}$ is our gluing operator from Definition 1.

For a constant number of colors $c_{\max}$, the set of size-$(t+1)$ parsing operators is finite. Adapting the proof of Downey and Fellows [6, Theorem 6.72], we verify that the graphs generated by the operators in Definition 2 have treewidth at most t. The same proof shows how, from a width-t tree decomposition of a colored graph G with at least $t+1$ vertices, a linear-size parse tree over the above operators can be obtained in linear time that generates a graph isomorphic to G.

Definition 3. The set $\mathcal{U}^{\text{small}}_{t,c_{\max}}$ is the set of $\{1, \ldots, c_{\max}\}$-colored t-boundaried graphs that can be generated by the operators in Definition 2.

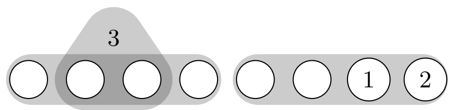
Theorem 1. *Let $F \subseteq \mathcal{U}^{small}_{t,c_{\max}}$ be a family of graphs. The following statements are equivalent:*

i) The parse trees corresponding to graphs in F are recognizable by a finite tree automaton.
ii) The canonical right congruence $\sim_F$ has finite index over $\mathcal{U}^{small}_{t,c_{\max}}$.

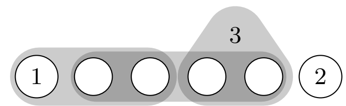
2.2 Lifting to Hypergraphs

To make the method accessible to hypergraph problems, we lift the Myhill-Nerode theorem for colored graphs of the previous subsection to hypergraphs.

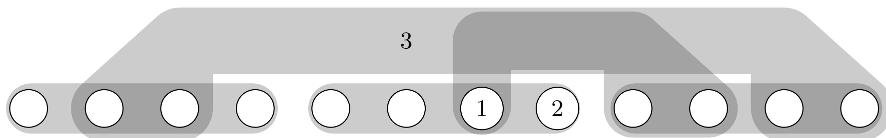
Definition 4. A t-boundaried hypergraph G has t distinguished vertices and hyperedges labeled from 1 to t. Two t-boundaried hypergraphs are *gluable* if no vertex of one hypergraph has the label of a hyperedge of the other hypergraph.



(a) A 3-boundaried hypergraph $\mathcal{H}(G)$.



(b) A 3-boundaried hypergraph $\mathcal{H}(H)$.



(c) The glued hypergraph $\mathcal{H}(G) \oplus_{\mathrm{h}} \mathcal{H}(H) = \mathcal{H}(G \oplus_{\mathrm{c}} H)$.

Fig. 2. The two hypergraphs represented by the well-colored 3-boundaried graphs G and H in Figure 1 and the glued hypergraph $\mathcal{H}(G) \oplus_{\mathrm{h}} \mathcal{H}(H) = \mathcal{H}(G \oplus_{\mathrm{c}} H)$.

Let G_1 and G_2 be gluable t-boundaried hypergraphs. We denote by $G_1 \oplus_{\mathrm{h}} G_2$ the t-boundaried hypergraph obtained by taking the disjoint union of G_1 and G_2, identifying each labeled vertex of G_1 with the vertex of G_2 with the same label, and replacing the hyperedges with label ℓ by the union of these hyperedges.

Let $\mathcal{H}_t^{\mathrm{large}}$ be the universe of t-boundaried hypergraphs and $F \subseteq \mathcal{H}_t^{\mathrm{large}}$. We define the *canonical right congruence* $\sim_F$ for F as follows: for two gluable $G_1, G_2 \in \mathcal{H}_t^{\mathrm{large}}$, $G_1 \sim_F G_2$ if and only if for all $H \in \mathcal{H}_t^{\mathrm{large}}$ that are gluable to G_1 and G_2, $G_1 \oplus_{\mathrm{h}} H \in F \iff G_2 \oplus_{\mathrm{h}} H \in F$.

To prove a Myhill-Nerode theorem for hypergraphs, we still need a way to create hypergraphs from parsing operators, as we did using the operators from Definition 2 for colored graphs. To this end, we indeed simply use the operators from Definition 2, observing that every bipartite graph can be interpreted as the incidence graph of a hypergraph. Moreover, if the two disjoint independent sets of a two-colored bipartite graph have distinct colors, then we can interpret this bipartite graph as a hypergraph in a unique way.

Definition 5. A *well-colored t-boundaried graph* is a $\{1,2\}$-colored t-boundaried graph $G = (U \uplus W, E)$, where the vertices in U have color 1, the vertices in W have color 2, and where U and W are independent sets.

For a well-colored t-boundaried graph $G = (U \uplus W, E)$, we denote by $\mathcal{H}(G)$ the t-boundaried hypergraph with the vertex set U and the edge set $\{N(w) : w \in W\}$. Moreover, vertices of $\mathcal{H}(G)$ inherit their label from G and edges $e = N(w), w \in W$ of $\mathcal{H}(G)$ inherit the label of w. For a set $U \subseteq \mathcal{U}_{t,2}^{\mathrm{large}}$, we denote $\mathcal{H}(U) := \{H \in \mathcal{H}_t^{\mathrm{large}} \mid H = \mathcal{H}(G), G \in U\}$.

The *incidence graph* of $\mathcal{H}(G)$ is G and the *incidence treewidth* of $\mathcal{H}(G)$ is the treewidth of G.

Obviously, every $H \in \mathcal{H}_t^{\mathrm{large}}$ is $\mathcal{H}(G)$ for some $G \in \mathcal{U}_{t,2}^{\mathrm{large}}$. Also, every set $\mathcal{F} \subseteq \mathcal{H}_t^{\mathrm{large}}$ is $\mathcal{F} = \mathcal{H}(F)$ for some $F \subseteq \mathcal{U}_{t,2}^{\mathrm{large}}$. Figure 2 illustrates Definitions 4 and 5.

Theorem 2. *Let $\mathcal{F} \subseteq \mathcal{U}_{t,2}^{small}$ be a set of well-colored t-boundaried graphs. The following statements are equivalent:*

i) The parse trees corresponding to graphs in F are recognizable by a finite tree automaton.
ii) The canonical right congruence $\sim_{\mathcal{H}(F)}$ has finite index over $\mathcal{H}(\mathcal{U}_{t,2}^{small})$.

We can use Theorem 2 to constructively derive algorithms for deciding properties of hypergraphs of incidence treewidth at most t: assume that we have a family of t-boundaried hypergraphs $\mathcal{F} \subseteq \mathcal{H}_t^{\text{large}}$ with incidence treewidth at most t such that $\sim_{\mathcal{F}}$ has finite index over $\mathcal{H}_t^{\text{large}}$. There is a set $F \subseteq \mathcal{U}_{t,2}^{\text{large}}$ for which $\mathcal{F} = \mathcal{H}(F)$.

We can decide in linear time whether a given hypergraph $H \in \mathcal{H}_t^{\text{large}}$ is isomorphic to a hypergraph in $\mathcal{F}$: compute the incidence graph G of H, that is, $\mathcal{H}(G) = H$. Since the graph G has treewidth at most t, we can compute a tree decomposition for G in linear time [2]. In the same way as shown by Downey and Fellows [6, Theorem 6.72], this tree decomposition can be converted in linear time into a parse tree T over the operators in Definition 2 that generates a graph isomorphic to G. By Theorem 2, a finite tree automaton can check whether T generates a graph G' isomorphic to a graph in F, which is the case if and only if $\mathcal{H}(G')$ is isomorphic to some graph in $\mathcal{F}$. The finite tree automaton can be constructed from the equivalence classes of $\sim_{\mathcal{F}}$ in constant time, since each equivalence class has a constant-size representative.

3 Hypergraph Cutwidth is fixed-parameter tractable

In this section we show that Hypergraph Cutwidth is fixed-parameter linear. We first formally define the problem.

Let $H = (V, E)$ be a hypergraph. A *linear layout* of H is an injective map $l\colon V \to \mathbb{R}$ of vertices onto the real line. The *cut at position* $i \in \mathbb{R}$ with respect to l, denoted $\theta_{l,H}(i)$, is the set of hyperedges that contain at least two vertices v, w such that $l(v) < i < l(w)$. The *cutwidth* of the layout l is $\max_{i\in\mathbb{R}} |\theta_{l,H}(i)|$. The *cutwidth* of the hypergraph H is the minimum cutwidth over all the linear layouts of H. The Hypergraph Cutwidth problem is then defined as follows.

Hypergraph Cutwidth
Input: A hypergraph $H = (V, E)$ and a natural number k.
Question: Does H have cutwidth at most k?

Now, to solve Hypergraph Cutwidth using our Myhill-Nerode analog, let k-HCW denote the class of all hypergraphs with cutwidth at most k. We will use Theorem 2 to show that the parse trees for graphs in k-HCW can be recognized by a finite tree automaton. To make Theorem 2 applicable, we first show that, for the hypergraphs in k-HCW, we can find a constant upper bound t on their incidence treewidth. This implies that isomorphic graphs can be generated by linear-size parse trees over the operator set in Definition 2 or, in terms of Theorem 2, that the graphs in k-HCW are isomorphic to the graphs in a subset of $\mathcal{H}(\mathcal{U}_{t,2}^{\text{small}})$.

Lemma 1. *Let $H = (V, E)$ be a hypergraph. If H has cutwidth at most k, then H has incidence treewidth at most* $\max\{k, 1\}$.

It remains to prove that the canonical right congruence $\sim_{k\text{-HCW}}$ of k-HCW has finite index. This finally shows that k-HCW can be recognized in linear time and, therefore, that Hypergraph Cutwidth is fixed-parameter linear.

To show that $\sim_{k\text{-HCW}}$ has finite index, we show that, given a t-boundaried hypergraph G, only a finite number of bits of information about a t-boundaried hypergraph H is needed in order to decide whether $G \oplus_{\mathrm{h}} H \in k$-HCW. To this end, we employ the Method of Test Sets [6]: let $\mathcal{T}$ be a set of objects called *tests* (for the moment, it is not important what exactly a test is). A t-boundaried graph can *pass* a test. For t-boundaried hypergraphs G_1 and G_2, let $G_1 \sim_{\mathcal{T}} G_2$ if and only if G_1 and G_2 pass the same subset of tests in $\mathcal{T}$. Obviously, $\sim_{\mathcal{T}}$ is an equivalence relation. Our aim is to find a set $\mathcal{T}$ of tests such that $\sim_{\mathcal{T}}$ refines $\sim_{k\text{-HCW}}$ (that is, if $G_1 \sim_{\mathcal{T}} G_2$, then $G_1 \sim_{k\text{-HCW}} G_2$). This will imply that, if $\sim_{\mathcal{T}}$ has finite index, so has $\sim_{k\text{-HCW}}$. To show that $\sim_{\mathcal{T}}$ has finite index, we show that we can find a *finite* set $\mathcal{T}$ such that $\sim_{\mathcal{T}}$ refines $\sim_{k\text{-HCW}}$.

Intuitively, in our case, a hypergraph G will pass a test T if it has a restricted linear layout, where each of its boundary vertices gets mapped to predefined integer values and each of the remaining vertices "lands" within one of a set of given "landing zones" between the integer values of the real line. This restricted linear layout will impose the same restrictions on an optimal cutwidth layout for G that are also imposed by an optimal cutwidth layout of $G \oplus_{\mathrm{h}} H$ for some hypergraph H that corresponds to T.

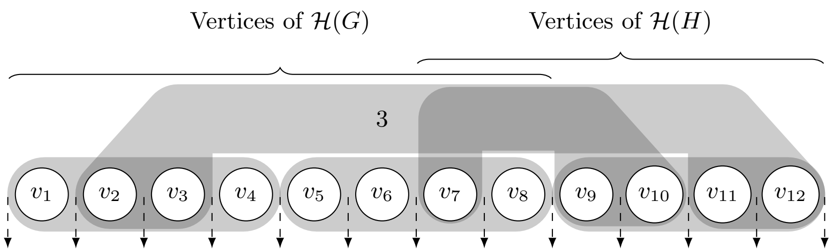
Definition 6. Let G and H be t-boundaried hypergraphs such that $G \oplus_{\mathrm{h}} H \in k$-HCW. Let v_i denote the vertex that is mapped to position i in an optimal cutwidth layout that maps to integer values.

A *landing zone* is a tuple in $\{0, \ldots, k\} \times 2^{\{1,\ldots,t\}}$. A *size-$n$ test* $T = (\pi, S)$ consists of a map $\pi\colon \{1, \ldots, t\} \to \{1, \ldots, n\}$ and a sequence $S = (S_0, S_1, \ldots, S_n)$ of landing zones.

We define an *H-test* $T = (\pi, S)$ as follows: for each vertex $v_i \in \partial(H)$, set $\pi(\ell) := i$, where ℓ is the label of v_i. For $i \in \{0, \ldots, n\}$, $S_i := (w_i, E_i)$, where

1. w_i is the number of hyperedges in H that contain vertices in $\{v_1, \ldots, v_i\} \cap V(H)$ and $\{v_{i+1}, \ldots, v_n\} \cap V(H)$.
2. E_i is the set of labels of hyperedges in H containing vertices in $\{v_1, \ldots, v_i\} \cap V(H)$ and $\{v_{i+1}, \ldots, v_n\} \cap V(H)$.

Figure 3 illustrates this definition. We now formally define what it means to pass a test. Intuitively, if a graph G passes an H-test, then $G \oplus_{\mathrm{h}} H \in k$-HCW.



	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}
w_i	0	0	0	0	0	0	0	1	1	2	1	2	0
E_i	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\{3\}$	$\emptyset$

Fig. 3. Construction of the H-test illustrated using the glued hypergraph $\mathcal{H}(G) \oplus_{\mathrm{h}} \mathcal{H}(H)$ from Figure 2.

Definition 7. Let $G = (V, E)$ be a t-boundaried hypergraph and $T = (\pi, S)$ be an H-test for some t-boundaried hypergraph H, where $S = (S_0, \dots, S_n)$ and $S_i = (w_i, E_i)$.

A *T-compatible layout for G* is an injective function $f \colon V \to \mathbb{R}$ such that each vertex $v \in \partial(G)$ with label ℓ is mapped to $\pi(\ell)$ and such that every vertex $v \in V \setminus \partial(G)$ is mapped into some open interval $(i, i+1)$ for $0 \le i \le n$.

The *weighted cutwidth of f* is $\max_{i \in \mathbb{R}}(|\theta_f(i)| + w_{\lfloor i \rfloor})$, where $\theta_f(i)$ is the set of hyperedges containing two vertices v and w with $f(v) < i < f(w)$ and that do not have a label in $E_{\lfloor i \rfloor}$.

Finally, *G passes the test T* if there is a T-compatible layout f for G whose weighted cutwidth is at most k.

Lemma 2. *For $\mathcal{T}$ being the set of all tests, the equivalence relation $\sim_{\mathcal{T}}$ refines $\sim_{k\text{-HCW}}$.*

Proof (Sketch). We show that if two t-boundaried hypergraphs G_1, G_2 pass the same subset of tests of $\mathcal{T}$, then, for all t-boundaried hypergraphs H, $G_1 \oplus_{\mathrm{h}} H \in k$-HCW if and only if $G_2 \oplus_{\mathrm{h}} H \in k$-HCW. We exploit the following two claims.

1. If $G_1 \oplus_{\mathrm{h}} H \in k$-HCW, then G_1 passes the H-test.
2. If G_2 passes the H-test, then $G_2 \oplus_{\mathrm{h}} H \in k$-HCW.

Let H be a t-boundaried hypergraph such that $G_1 \oplus_{\mathrm{h}} H \in k$-HCW, and let T be an H-test. By (1), G_1 passes T. Since G_1 and G_2 pass the same tests, also G_2 passes T. By (2), it follows that $G_2 \oplus_{\mathrm{h}} H \in k$-HCW. The reverse direction is proved symmetrically. See the appendix for the proofs of Claims (1) and (2). $\square$

Lemma 2 shows a set of tests $\mathcal{T}$ such that $\sim_{\mathcal{T}}$ refines $\sim_{k\text{-HCW}}$. However, the set $\mathcal{T}$ is infinite and, therefore, does not yet yield that $\sim_{k\text{-HCW}}$ has finite index. However, we can obtain a finite set of tests using the following lemma.

Lemma 3. *Let G and H be t-boundaried hypergraphs. For every H-test T_1, there is a test T_2 of size $2t(2k+2)$ such that G passes T_1 if and only if it passes T_2.*

Now the following theorem is easy to prove.

Theorem 3. Hypergraph Cutwidth *is fixed-parameter linear.*

4 Hypertree Width and Variants

In this section we sketch a negative application of our hypergraph Myhill-Nerode analog to Generalized Hypertree Width [12]. The problem is, given a hypergraph H and an integer k as input, to decide whether H has generalized hypertree width at most k. Since Generalized Hypertree Width is NP-hard for $k = 3$ [12], it would be nice to find parameters for which the problem is fixed-parameter tractable. However, we can show that Generalized Hypertree Width does not have finite index.

Theorem 4. *Let $k \geq 0$. Let k-GHTW be the family of all incidence graphs G such that $\mathcal{H}(G)$ has generalized hypertree width at most k. The canonical right congruence $\sim_{\mathcal{H}(k\text{-GHTW})}$ does not have finite index over $\mathcal{H}(\mathcal{U}_{t,2}^{small})$.*

Moreover, the construction we use in the proof leads us to conjecture the problem to be W[1]-hard with respect to the parameter incidence treewidth. The proof also applies to the problem variants Hypertree Width and Fractional Hypertree Width, which are unlikely to be fixed-parameter tractable with respect to their standard parameterization [11, 16].

Conjecture 1. Hypertree Width is W[1]-hard with respect to the parameter incidence treewidth.

5 Conclusion

We have extended the graph analog of the Myhill-Nerode theorem to colored graphs and hypergraphs, making the methodology more widely applicable. Our positive application shows that Hypergraph Cutwidth is fixed-parameter linear. As a negative application, we showed that Hypertree Width, Generalized Hypertree Width, and Fractional Hypertree Width do not have finite index, and therefore the parse trees associated to Yes-instances of bounded incidence treewidth cannot be recognized by finite tree automata.

Acknowledgments. The authors are thankful to Mahdi Parsa for fruitful discussions. René van Bevern acknowledges support by the DFG, project DAPA (NI 369/12-1). The other authors acknowledge support by by the Australian Research Council, grants DP 1097129 (Michael R. Fellows), DE 120101761 (Serge Gaspers), and DP 110101792 (Michael R. Fellows and Frances A. Rosamond). NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

Bibliography

[1] K. R. Abrahamson and M. R. Fellows. Finite automata, bounded treewidth, and well-quasiordering. In *Graph Structure Theory*, volume 147 of *Contemporary Mathematics*, pages 539–564. American Mathematical Society, 1991.

[2] H. L. Bodlaender. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.*, 25(6):1305–1317, 1996.

[3] H. L. Bodlaender, M. R. Fellows, and D. M. Thilikos. Derivation of algorithms for cutwidth and related graph layout parameters. *J. Comput. Syst. Sci.*, 75(4): 231–244, 2009.

[4] J. Cahoon and S. Sahni. Exact algorithms for special cases of the board permutation problem. In *Proceedings of the 21st Annual Allerton Conference on Communication, Control, and Computing*, pages 246–255, 1983.

[5] R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer Verlag, New York, 4th edition, 2010.

[6] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer, 1999.

[7] M. R. Fellows and M. A. Langston. An analogue of the Myhill-Nerode Theorem and its use in computing finite-basis characterizations (extended abstract). In *Proc. 30th FOCS*, pages 520–525. IEEE Computer Society, 1989.

[8] M. R. Fellows and M. A. Langston. On well-partial-order theory and its application to combinatorial problems of VLSI design. *SIAM J. Discrete Math.*, 5(1):117–126, 1992.

[9] M. R. Fellows and M. A. Langston. On search, decision, and the efficiency of polynomial-time algorithms. *J. Comput. Syst. Sci.*, 49(3):769–779, 1994.

[10] F. Gavril. Some NP-complete problems on graphs. In *Proc. 1977 Conf. on Inf. Sc. and Syst.*, pages 91–95. Johns Hopkins Univ., 1977.

[11] G. Gottlob, M. Grohe, N. Musliu, M. Samer, and F. Scarcello. Hypertree decompositions: Structure, algorithms, and applications. In *Proc. 31st WG*, volume 3787 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2005.

[12] G. Gottlob, Z. Miklós, and T. Schwentick. Generalized hypertree decompositions: NP-hardness and tractable variants. *J. ACM*, 56(6), 2009.

[13] P. Hliněný. On matroid properties definable in the MSO logic. In *Proc. 28th MFCS*, volume 2747 of *LNCS*, pages 470–479. Springer, 2003.

[14] N. Kashyap. Matroid pathwidth and code trellis complexity. *SIAM J. Discret. Math.*, 22(1):256–272, 2008.

[15] P. G. Kolaitis and M. Y. Vardi. Conjunctive-query containment and constraint satisfaction. *J. Comput. Sci.*, 61(2):302–332, 2000.

[16] D. Marx. Approximating fractional hypertree width. *ACM Transactions on Algorithms*, 6(2), 2010.

[17] Z. Miller and I. H. Sudborough. A polynomial algorithm for recognizing bounded cutwidth in hypergraphs. *Mathematical Systems Theory*, 24(1):11–40, 1991.

[18] H. Nagamochi. Linear layouts in submodular systems. In *Proc. 23rd ISAAC*, volume 7676 of *LNCS*, pages 475–484. Springer, 2012. ISBN 978-3-642-35260-7.

[19] M. Samer and S. Szeider. Constraint satisfaction with bounded treewidth revisited. *J. Comput. Sci.*, 76(2):103–114, 2010.

[20] D. M. Thilikos, M. J. Serna, and H. L. Bodlaender. Cutwidth I: A linear time fixed parameter algorithm. *J. Algorithms*, 56(1):1–24, 2005.

[21] A. Vardy. Trellis structure of codes. In *Handbook of Coding Theory*, chapter 24. Elsevier, 1998.

A Proofs of Section 2

A.1 Proof of Theorem 1

Two colored t-boundaried graphs G_1 and G_2 are *isomorphic* and we write $G_1 \cong G_2$ if there is a graph isomorphism for the underlying (ordinary) graphs mapping each vertex to a vertex with the same color.

Our proof of Theorem 1 adapts the proof of Downey and Fellows [6, Theorem 6.77]. To this end, we only need to show that our operators satisfy the Parsing Replacement Property.

Definition 8. *(Parsing Replacement Property).* An n-ary operator $\otimes$ has the *Parsing Replacement Property* if, for the arguments $H_1, \dots, H_n \in \mathcal{U}_{t,c_{\max}}^{\text{small}}$ of $\otimes$ and each i, there is a graph $G \in \mathcal{U}_{t,c_{\max}}^{\text{small}}$ such that $H_1 \otimes \dots \otimes H_n \cong H_i \oplus_{\text{c}} G$.

Lemma 4. *The size-$(t+1)$ parsing operators in Definition 2 have the Parsing Replacement Property.*

Proof. Let $H \in \mathcal{U}_{t,c_{\max}}^{\text{small}}$ and let $\emptyset_H^*$ denote the graph that contains only $t+1$ boundary vertices, no edges, and that is color-compatible with H. Finally, let $\emptyset_H^\ell$ denote the graph that contains $t+1$ boundary vertices and one additional vertex with color ℓ and that is color-compatible with H. Note that $\emptyset_H^\ell, \emptyset_H^* \in \mathcal{U}_{t,c_{\max}}^{\text{small}}$: we can generate them using the operators in Definition 2 because $\emptyset_H^*$ and $\emptyset_H^\ell$ have at least $t+1$ vertices and treewidth at most t. Now, the lemma immediately follows, noting that for each graph $H \in \mathcal{U}_{t,c_{\max}}^{\text{small}}$,

$$\begin{aligned} \gamma H \cong iH &\cong H \oplus_{\text{c}} \emptyset_H^*, \text{ since } \cong \text{ ignores boundary labels,} \\ eH &\cong H \oplus_{\text{c}} e(\emptyset_H^*), \\ u_\ell H &\cong H \oplus_{\text{c}} \emptyset_H^\ell, \end{aligned}$$

and that $\oplus_{\text{c}}$ and $\emptyset_{n_1,\dots,n_{c_{\max}}}$ trivially have the Parsing Replacement Property. □

Having proven Lemma 4, the proof of [6, Theorem 6.77] also proves Theorem 1 and we have a handy tool to generalize the Myhill-Nerode theorem to hypergraphs.

A.2 Proof of Theorem 2

Theorem 2. Let $\mathcal{F} \subseteq \mathcal{U}_{t,2}^{\text{small}}$ be a set of well-colored t-boundaried graphs. The following statements are equivalent:

i) The parse trees corresponding to graphs in F are recognizable by a finite tree automaton.
ii) The canonical right congruence $\sim_{\mathcal{H}(F)}$ has finite index over $\mathcal{H}(\mathcal{U}_{t,2}^{\text{small}})$.

Proof. Let $\bar{F} := \{G \in \mathcal{U}_{t,2}^{\text{small}} : \mathcal{H}(G) \in \mathcal{H}(F)\}$, that is, for each $G \in F$, the set $\bar{F}$ contains additionally all t-boundaried graphs $G' \in \mathcal{U}_{t,2}^{\text{small}}$ for which $H(G) = H(G')$. Clearly, every such G' is isomorphic to G, that is, $\bar{F}$ contains F and t-boundaried graphs isomorphic to graphs in F. Hence, the parse trees corresponding to t-boundaried graphs in F are the same as those corresponding to $\bar{F}$. Moreover, $\mathcal{H}(F) = \mathcal{H}(\bar{F})$. Hence, it remains to show that $\sim_{\mathcal{H}(\bar{F})}$ has finite index over $\mathcal{H}(\mathcal{U}_{t,2}^{\text{small}})$ if and only if $\sim_{\bar{F}}$ has finite index over $\mathcal{U}_{t,2}^{\text{small}}$.

Assume that $\sim_{\bar{F}}$ has infinite index over $\mathcal{U}_{t,2}^{\text{small}}$. Then there is a family $G_1, G_2, G_3, \dots$ of graphs in $\mathcal{U}_{t,2}^{\text{small}}$ that are pairwise inequivalent under $\sim_{\bar{F}}$. Since there are only a finite number of possibilities to assign two colors to t boundary vertices, there are an infinite number of color-compatible graphs among $G_1, G_2, \dots$. Moreover, notice that all G_i that are not well-colored t-boundaried graphs are equivalent under $\sim_{\bar{F}}$ (they cannot be completed to graphs in $\bar{F}$ by gluing any graph onto them). Therefore, without loss of generality, we assume that $G_1, G_2, \dots$ are pairwise color-compatible well-colored t-boundaried graphs. Now, for each pair G_i, G_j, there is a graph $H_{i,j} \in \mathcal{U}_{t,2}^{\text{small}}$ such that $G_i \oplus_{\text{c}} H_{i,j} \in \bar{F}$ but $G_j \oplus_{\text{c}} H_{i,j} \notin \bar{F}$. From $G_i \oplus_{\text{c}} H_{i,j} \in \bar{F}$, it follows that $H_{i,j}$ is a well-colored t-boundaried graph that is color-compatible with G_i. Now, we have $\mathcal{H}(G_i) \oplus_{\text{h}} \mathcal{H}(H_{i,j}) = \mathcal{H}(G_i \oplus_{\text{c}} H_{i,j}) \in \mathcal{H}(\bar{F})$. Moreover, $\mathcal{H}(G_j) \oplus_{\text{h}} \mathcal{H}(H_{i,j}) = \mathcal{H}(G_j \oplus_{\text{c}} H_{i,j}) \notin \mathcal{H}(\bar{F})$. That is, $\mathcal{H}(G_i) \nsim_{\mathcal{H}(\bar{F})} \mathcal{H}(G_j)$ and therefore $\sim_{\mathcal{H}(\bar{F})}$ has infinite index.

Assume that $\sim_{\mathcal{H}(\bar{F})}$ has infinite index over $\mathcal{H}(\mathcal{U}_{t,2}^{\text{small}})$. Then, there is a family $\mathcal{H}(G_1), \mathcal{H}(G_2), \mathcal{H}(G_3), \dots$ of hypergraphs in $\mathcal{H}(\mathcal{U}_{t,2}^{\text{small}})$ that are pairwise inequivalent under $\sim_{\bar{F}}$. Since there are only a finite number of partitions of t labels into hyperedge-labels and vertex-labels, there is an infinite number of pairwise gluable hypergraphs among $\mathcal{H}(G_1), \mathcal{H}(G_2), \dots$. Therefore, without loss of generality, assume that all these hypergraphs are pairwise gluable. Now, for each pair $\mathcal{H}(G_i)$, $\mathcal{H}(G_j)$, there is a hypergraph $\mathcal{H}(H_{i,j}) \in \mathcal{H}(\mathcal{U}_{t,2}^{\text{small}})$ such that $\mathcal{H}(G_i) \oplus_{\text{h}} \mathcal{H}(H_{i,j}) \in \mathcal{H}(\bar{F})$ but $\mathcal{H}(G_j) \oplus_{\text{h}} \mathcal{H}(H_{i,j}) \notin \mathcal{H}(\bar{F})$. Hence, since $\mathcal{H}(G_i) \oplus_{\text{h}} \mathcal{H}(H_{i,j}) = \mathcal{H}(G_i \oplus_{\text{c}} H_{i,j})$, we have $G_i \oplus_{\text{c}} H_{i,j} \in \bar{F}$. However, with the same reasoning, $G_j \oplus_{\text{c}} H_{i,j} \notin \bar{F}$. Hence, $\sim_{\bar{F}}$ has infinite index over $\bar{F}$. ◻

B Proofs of Section 3

B.1 Proof of Lemma 1

Lemma 1. Let $H = (V, E)$ be a hypergraph. If H has cutwidth at most k, then H has incidence treewidth at most $\max\{k, 1\}$.

Proof. Suppose H has cutwidth at most k. Let $H' = (V, E')$ denote the hypergraph obtained from H by removing all hyperedges of size at most 1. Consider a linear layout l of cutwidth at most k of the vertices of H'. Without loss of generality, assume that l maps to the natural numbers 1 to n and let $V = \{v_1, \dots, v_n\}$ be such that $l(v_i) = i$. We construct a path decomposition for the incidence graph $\mathcal{I}(H')$ with the bags $L_1, R_1, L_2, R_2, \ldots, L_{n-1}, R_{n-1}$ and a path connecting the bags in this order. For every $i, 1 \leq i \leq n-1$, let $L_i := \theta_{l,H'}(i) \cup \{v_i\}$ and $R_i := \theta_{l,H'}(i) \cup \{v_{i+1}\}$, recalling that $\theta_{l,H'}(i)$ is the set of hyperedges that cross position i.

We now prove that this is a path decomposition for $\mathcal{I}(H')$. Let $\{v_i, e\}$ be any edge in the incidence graph of H' with $v_i \in V$ and $e \in E'$. Since $|e| \geq 2$, we have that either $e \in \theta_{l,H'}(i-1)$ or $e \in \theta_{l,H'}(i)$. Therefore, either $e \in R_{i-1}$ or $e \in L_i$. But $v_i \in R_{i-1} \cap L_i$. Thus, v_i and e occur together in at least one bag. Next, consider the vertex $v_i \in V$. It occurs in the two bags R_{i-1} and L_i, which are consecutive and thus induce a connected path. Finally, consider a hyperedge $e \in E'$. It occurs in all bags $L_i, R_i, \ldots, L_j, R_j$ where v_i is the first vertex in the layout l occurring in e and v_j is the last vertex in l occurring in e. But these bags are all consecutive on the path and thus induce a connected path.

The width of this path decomposition is $\max_{1 \leq i \leq n-1}\{|\theta_{H',l}(i)|\} \leq k$ by construction. To obtain a tree decomposition for H from this path decomposition for H', we only need to take care of hyperedges of size at most 1. For every hyperedge $e \in E$ of size 1, add a new bag $\{e, v\}$, where v is the unique vertex contained in e, and make it adjacent to an arbitrary bag containing v. For every hyperedge $e \in E$ of size 0, add a new bag $\{e\}$, and make it adjacent to an arbitrary bag. In this way, we obtain a tree decomposition for the incidence graph of H of width at most $\max\{k, 1\}$. Thus, H has incidence treewidth at most $\max\{k, 1\}$. □

B.2 Proof of Lemma 2

Lemma 2. For $\mathcal{T}$ being the set of all tests, the equivalence relation $\sim_{\mathcal{T}}$ refines $\sim_{k\text{-HCW}}$.

Proof. We show that if two t-boundaried hypergraphs G_1, G_2 pass the same subset of tests of $\mathcal{T}$, then, for all t-boundaried hypergraphs H, $G_1 \oplus_{\mathrm{h}} H \in k$-HCW if and only if $G_2 \oplus_{\mathrm{h}} H \in k$-HCW. The proof is based on the following two claims that are proved afterwards.

i) If $G_1 \oplus_{\mathrm{h}} H \in k$-HCW, then G_1 passes the H-test.
ii) If G_2 passes the H-test, then $G_2 \oplus_{\mathrm{h}} H \in k$-HCW.

Let H be a t-boundaried hypergraph such that $G_1 \oplus_{\mathrm{h}} H \in k$-HCW, and let T be an H-test. By (i), G_1 passes T. Since G_1 and G_2 pass the same tests, also G_2 passes T. By (ii), it follows that $G_2 \oplus_{\mathrm{h}} H \in k$-HCW. The reverse direction is proved symmetrically. It only remains to prove (i) and (ii).

(i) Let l be a layout of $G_1 \oplus_{\mathrm{h}} H$ with cutwidth at most k that only maps boundary vertices to integral positions and let T be the H-test. The restriction f of l to the vertex set of G_1 is then clearly T-compatible. Moreover, f has weighted cutwidth at most k: to this end, for an arbitrary $i \in \mathbb{N}$, we show that $|\theta_f(i)| + w_{\lfloor i \rfloor}$ from Definition 7 is at most k. Consider the set A of hyperedges of $G_1 \oplus_{\mathrm{h}} H$ that contain two vertices v, w with $f(v) < i < f(w)$. Obviously, $|A| \leq k$. We show $|\theta_f(i)| + w_{\lfloor i \rfloor} = |A|$. The set A can be partitioned as $A = A^H \uplus A^{G_1} \uplus A^*$, where

A^H are hyperedges in H, A^{G_1} are hyperedges in G_1 and A^* are hyperedges that are neither in H nor in G_1. The hyperedges in A^* are labeled edges that are the result of taking the unions of equally-labeled hyperedges from G_1 and H. We further partition $A^* = A^+ \uplus A^-$ such that A^+ is the set of hyperedges with labels in $E_{\lfloor i \rfloor}$. Now, by definition, we have $w_{\lfloor i \rfloor} = |A^H|+|A^+|$ and $|\theta_f(i)| = |A^{G_1}|+|A^-|$. Therefore, $|\theta_f(i)| + w_{\lfloor i \rfloor} = |A| \le k$.

Note here, that, if we were to define $\theta_f(i)$ in Definition 7 to contain *all* edges that contain vertices left and right of i (rather than only those with no label in $E_{\lfloor i \rfloor}$), then we would obtain $|\theta_f(i)| = |A^{G_1}| + |A^+| + |A^-|$ here, that is, $|\theta_f(i)| + w_{\lfloor i \rfloor}$ would count the edges in A^+ twice.

(ii) The H-test T was obtained from a linear layout l of cutwidth k for $G_1 \oplus_{\mathrm{h}} H$. Moreover, there is a T-compatible layout f for G_2. First note that l and f agree on the layout of vertices in $\partial(G_2)$ and $\partial(H)$ and that, apart from these, f lays out vertices at non-integral positions, whereas l lays out vertices at integral positions. Because of this, in a layout g for $G_2 \oplus_{\mathrm{h}} H$ that lays out vertices v of H at position $l(v)$ and vertices v of G_2 at position $f(v)$, every two vertices in $G_2 \oplus_{\mathrm{h}} H$ are laid out at distinct positions by g. Hence, g is injective and, therefore, a layout.

Now it is easy to observe that g is a layout of cutwidth at most k for $G_2 \oplus_{\mathrm{h}} H$: to this end, for a position $i \in \mathbb{R}$, consider the set A of hyperedges of $G_2 \oplus_{\mathrm{h}} H$ containing vertices v, w with $g(v) < i < g(w)$ and let it be partitioned $A = A^H \uplus A^{G_2} \uplus A^+ \uplus A^-$ in the same way as above. Again, we have $w_{\lfloor i \rfloor} = |A^H|+|A^+|$ and $|\theta_f(i)| = |A^{G_2}| + |A^-|$, yielding $|A| = |A^{G_2}| + |A^H| + |A^+| + |A^-| = \theta_f(i) + w_{\lfloor i \rfloor}$. Since G_2 passes the H-test, this is at most k. □

B.3 Proof of Lemma 3

Lemma 3. Let G and H be t-boundaried hypergraphs. For every H-test T_1, there is a test T_2 of size $2t(2k+2)$ such that G passes T_1 if and only if it passes T_2.

Proof. Let the sequence S of landing zones of T_1 be $S_0 = (E_0, w_0), \dots, (E_n, w_n) = S_n$. For $E \subseteq \{1, \dots, t\}$, we call a maximal subsequence $(E_i, w_i), \dots, (E_j, w_j)$ of S with $E_i = \dots = E_j$ a *strait*. We first show that there are at most $2t$ straits, and then show that we can shorten each strait to a length of $2k+2$ by removing some landing zones without changing the satisfiability of the test.

For $\ell \subseteq \{1, \dots, t\}$, let $I_\ell := \{i \le n : \ell \in E_i\}$. Observe that I_ℓ is a consecutive subset of $\{0, \dots, n\}$: this is by the construction of the E_i in Definition 7 from an optimal layout for some t-boundaried graph $G' \oplus_{\mathrm{h}} H$. That is, each I_ℓ for some $\ell \in \{1, \dots, t\}$ is an interval of the natural numbers with a minimum element and a maximum element, which we both call *events*. Hence, the I_ℓ for all $\ell \in \{1, \dots, t\}$ in total have at most $2t$ events. Since straits can only start at an event, and since only one strait can start at a fixed event, it follows that S is partitioned into at most $2t$ straits.

It remains to shorten the straits. Let $(E_i, w_i), \dots, (E_j, w_j)$ be a strait. We apply the data reduction rules (R1–R3) from the proof of Theorem 6.82 by

Downey and Fellows [6], which are based on the following observations: deleting from S one of two consecutive landing zones $(E_i, w_i), (E_{i+1}, w_{i+1})$ with $E_i = E_{i+1}$ and $w_i = w_{i+1}$ yields an equivalent test, as does adding such landing zones. Moreover, if we replace a landing zone (E_i, w_i) by a landing zone (E_i, w_i') with $w_i' \leq w_i$, we obtain a test that is easier to pass.

Since the w_i are, by construction of T_1, bounded from above by k, we can transform T_1 into T_2 by these data reduction rules so that each of the $2t$ straits has length at most $(2k+2)$ [6, Theorem 6.83]. □

B.4 Proof of Theorem 3

Theorem 3. Hypergraph Cutwidth is fixed-parameter linear.

Proof. Lemma 1 shows that graphs in k-HCW have constant treewidth at most t, and therefore, that all such hypergraphs can be linear-time transformed into linear-size parse trees for t-boundaried hypergraphs. Using Theorem 2, we show that parse trees corresponding to hypergraphs in k-HCW are recognizable by a finite tree automaton: let $\mathcal{T}$ be the set of all tests and $\mathcal{T}'$ be the set of all tests of size $2t(2k+2)$. Lemma 3 shows that $\sim_{\mathcal{T}'}$ refines $\sim_{\mathcal{T}}$. Lemma 2 shows that $\sim_{\mathcal{T}}$ refines $\sim_{k\text{-HCW}}$. Therefore, $\sim_{k\text{-HCW}}$ has at most as many equivalence classes as $\sim_{\mathcal{T}'}$. Since k and t are constant, $\mathcal{T}'$ is finite, implying finite index for $\sim_{\mathcal{T}'}$ and, consequently, for $\sim_{k\text{-HCW}}$. □

C Proofs of Section 4

C.1 Proof of Theorem 4

Theorem 4. Let $k \geq 0$. Let k-GHTW be the family of all incidence graphs G such that $\mathcal{H}(G)$ has generalized hypertree width at most k. The canonical right congruence $\sim_{\mathcal{H}(k\text{-GHTW})}$ does not have finite index over $\mathcal{H}(\mathcal{U}_{t,2}^{\text{small}})$.

We give a construction of a t-boundaried hypergraph H_n with bounded incidence treewidth, for every $n \geq 1$. Then we show that $H_n \oplus_{\text{h}} H_m$ has generalized hypertree width 4 if and only if $n = m$. This implies an infinite number of equivalence classes for the canonical right congruence $\sim_{\mathcal{H}(4\text{-GHTW})}$.

For every $n \geq 1$, we construct a t-boundaried hypergraph H_n with $t = 28$, generalized hypertree width 4, and incidence treewidth at most 12. The vertex set of H_n is $V = A \cup B \cup C \cup D \cup S \cup T \cup X$, where $A = \{a, y\}, B = \{b, z\}, C = \{c, y\}, D = \{d, z\}, S = \{s_1, \ldots, s_8\}, T = \{t_1, \ldots, t_8\}$ and $X = \{x_1, \ldots, x_{6n}\}$. The hyperedge set of H_n is $E = \{A, B, C, D\} \cup B_S \cup \{S_c, S_d, S_y, S_z\} \cup B_T \cup \{T_a, T_b,$

$T_y, T_z\} \cup \{E_{3i}, E_{3i+1} : 1 \le i < 2n\} \cup \{E_{i,i+1} : 1 \le i < 6n\}$, where

$$
\begin{aligned}
B_S &\text{ is the set of all possible binary hyperedges on } S,\\
S_c &= \{c, s_1, s_2\}, S_d = \{d, s_3, s_4\},\\
S_y &= \{y, s_5, s_6\}, S_z = \{z, s_7, s_8\}\\
B_T &\text{ is the set of all possible binary hyperedges on } T,\\
T_a &= \{a, t_1, t_2\}, T_b = \{b, t_3, t_4\},\\
T_y &= \{y, t_5, t_6\}, T_z = \{z, t_7, t_8\}\\
E_1 &= \{s_8, x_1\},\\
E_{3i} &= \{a, c, y, x_{3i}\} \text{ for } 1 \le i < 2n,\\
E_{3i+1} &= \{b, d, z, x_{3i+1}\} \text{ for } 1 \le i < 2n,\\
E_{6n} &= \{x_{6n}, t_1\},\\
E_{6i+1,6i+2} &= \{a, b, x_{6i+1}, x_{6i+2}\} \text{ for } 0 \le i < n,\\
E_{6i+4,6i+5} &= \{c, d, x_{6i+4}, x_{6i+5}\} \text{ for } 0 \le i < n, \text{ and}\\
E_{3i,3i+1} &= \{x_{3i}, x_{3i+1}\} \text{ for } 1 \le i < 2n.
\end{aligned}
$$

The set of boundary hyperedges is $\{A, B, C, D, S_c, S_d, S_y, S_z, T_a, T_b, T_y, T_z\}$. The set of boundary vertices is $S \cup T$. They are labeled from 1 to 28 in this order and by increasing indices. See Figure 4 for an illustration of H_2 induced on $V \setminus (S \cup T)$.

In the construction of H_n, the vertices in S and T and the hyperedges containing them are only used to make sure that every tree decomposition of H_n with (generalized, fractional) hypertree width 4 contains a bag $\mathcal{B}_{-1}$ with the vertices $S \cup \{c, d, y\}$ and a bag $\mathcal{B}_{6n+1}$ with the vertices $T \cup \{b, y, z\}$. Since the sets $S \cup \{c, d, y, z\}$ and $T \cup \{a, b, y, z\}$ can also be covered by 4 hyperedges, all of which are boundary hyperedges, let $\mathcal{D} = (\{X_i : i \in I\}, T)$ be a tree decomposition for H_n with the bags $\mathcal{B}_{-1} = S \cup \{c, d, y, z\}$ and $\mathcal{B}_{6n+1} = T \cup \{a, b, y, z\}$. We observe that all other vertices of H_n occur in bags that are in the same connected component of the forest obtained from $\mathcal{D}$ by removing these two bags.

Claim 1. The tree decomposition $\mathcal{D}$ contains a bag $\mathcal{B}_i$, $0 \le i \le 6n$, with $\{s_8, x_1, c, d, y, z\} \subseteq B_0$, $\{t_1, x_{6n}, a, b, y, z\} \subseteq B_{6n}$, and $\{a, b, c, d, y, z, x_i, x_{i+1}\} \subseteq \mathcal{B}_i$, for every i, $1 \le i < 6n$.

Proof. The primal graph $\mathcal{G}(H_n)$ contains the cliques $\{s_8, x_1\}$, $\{a, b, x_1, x_2\}$, $\{a, b, x_2, x_3\}$, $\{a, c, y, x_3\}$, $\{x_3, x_4\}$, $\{x_4, b, d, z\}$, $\{c, d, x_4, x_5\}$, $\{c, d, x_5, x_6\}$, $\{a, c, y, x_6\}$, $\{x_6, x_7\}$, $\{b, d, z, x_7\}$, $\{a, b, x_7, x_8\}$, $\ldots$, $\{x_{6n}, t_1\}$, and every two consecutive cliques in this list intersect in at least one vertex. In particular, we observe the path $(s_8, x_1, x_2, \ldots, x_{6n}, t_1)$ in $\mathcal{G}(H_n)$. Thus, $\mathcal{D}$ contains bags $\mathcal{B}_0 \supseteq \{s_8, x_1\}$, $\mathcal{B}_{6n} \supseteq \{t_1, x_{6n}\}$, and $\mathcal{B}_i \supseteq \{x_i, x_{i+1}\}$, $1 \le i < 6n$. Moreover, each $\mathcal{B}_i$, $0 \le i < 6n$, contains c, d, y, z since $\mathcal{B}_{-1}$ contains c, d, y, z, $\mathcal{B}_{6n-1}$ contains c, d, $\mathcal{B}_{6n+1}$ contains y, z, and without loss of generality, we can assume the $\mathcal{B}_i$, $0 \le i \le 6n$, were chosen such that they are on the path from $\mathcal{B}_{-1}$ to $\mathcal{B}_{6n+1}$ in T. Similarly, each $\mathcal{B}_i$, $1 \le i \le 6n$, contains a, b. □

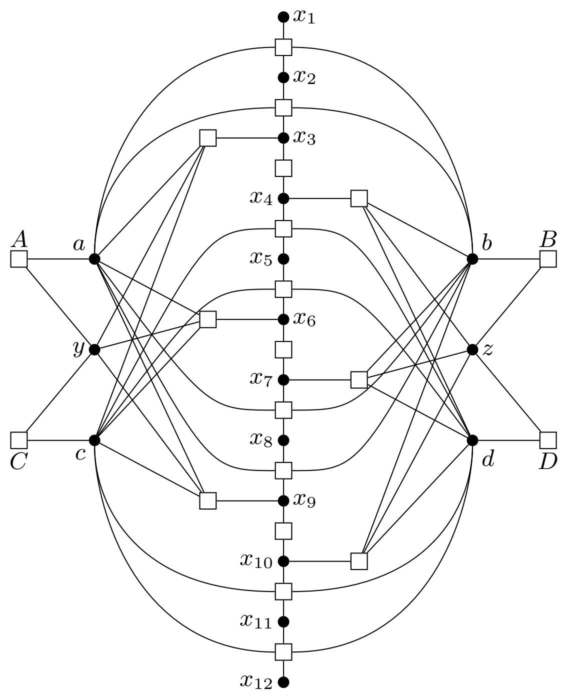


Fig. 4. The incidence graph of H_2 induced on $V \setminus (S \cup T)$. Boxes represent hyperedges.

A tree decomposition for H_n is a *good* tree decomposition if it contains the bags $\mathcal{B}_{-1} = S \cup \{c, d, y, z\}$ and $\mathcal{B}_{6n+1} = T \cup \{a, b, y, z\}$ and every bag except $\mathcal{B}_{-1}$ and $\mathcal{B}_{6n+1}$ can be covered with at most 3 hyperedges, and in case such a bag is covered with exactly 3 hyperedges, two of these hyperedges are in the boundary. A *good* cover for a good tree decomposition is a cover for each bag according to the specifications of a good tree decomposition.

Claim 2. If $\mathcal{D}$ is a good tree decomposition for H_n, then, for every i, $-1 \leq i \leq 6n$, there is a path from the bag $\mathcal{B}_i$ to the bag $\mathcal{B}_{i+1}$ that avoids all the bags $\mathcal{B}_j$, $j \in \{-1, \ldots, 6n+1\} \setminus \{i, i+1\}$.

Proof. Suppose the path from $\mathcal{B}_i$ to $\mathcal{B}_{i+1}$ passes through $\mathcal{B}_j$ with $j \in \{-1, \ldots, 6n+1\} \setminus \{i, i+1\}$. Since every bag on the path from $\mathcal{B}_i$ to $\mathcal{B}_{i+1}$ contains $\mathcal{B}_i \cap \mathcal{B}_{i+1}$, we have that $x_{i+1} \in \mathcal{B}_j$. But then $\{s_8, x_1, c, d, y, z, x_{i+1}\} \subseteq B_j$ (if $j = 0$) or $\{t_1, x_{6n}, a, b, y, z, x_{i+1}\} \subseteq B_j$ (if $j = 6n$) or $\{a, b, c, d, y, z, x_j, x_{j+1}, x_{i+1}\} \subseteq \mathcal{B}_j$ (otherwise), implying that B_j cannot be covered by two hyperedges and it cannot be covered by three hyperedges of which two are in the boundary. $\square$

Claim 3. In every good cover, $\mathcal{B}_0$ is covered by $\{E_1, C, D\}$, $\mathcal{B}_{6n}$ is covered by $\{E_{6n}, A, B\}$, and for every i, $1 \le i < 6n$,

$$\mathcal{B}_i \text{ is covered by } \begin{cases} \{E_{i,i+1}, C, D\} & \text{if } i \equiv 1 \pmod 6, \\ \{E_{i,i+1}, C, D\} & \text{if } i \equiv 2 \pmod 6, \\ \{E_i, E_{i+1}\} & \text{if } i \equiv 3 \pmod 6, \\ \{E_{i,i+1}, A, B\} & \text{if } i \equiv 4 \pmod 6, \\ \{E_{i,i+1}, A, B\} & \text{if } i \equiv 5 \pmod 6, \text{ and} \\ \{E_i, E_{i+1}\} & \text{if } i \equiv 0 \pmod 6. \end{cases}$$

Proof. The claim easily follows from Claim 1. □

Suppose $\mathcal{D}$ is a good tree decomposition for H_n. The *backbone* of $\mathcal{D}$ is the path P in T starting at the bag $\mathcal{B}_{-1}$ and ending at the bag $\mathcal{B}_{6n+1}$. By Claim 2, P visits $\mathcal{B}_0, \mathcal{B}_1, \ldots, \mathcal{B}_{6n}$ in this order. Let $P_{i,j}$ denote the subpath of P starting at $\mathcal{B}_i$ and ending at $\mathcal{B}_j$.

Claim 4. For every $i \in \{0, 6, 12, \cdots, 6n-6\}$, no bag on $P_{i,i+3}$ is covered by a set of hyperedges $\mathcal{Q}$ with $A, B \in \mathcal{Q}$ in a good cover.

Proof. Consider a bag $\mathcal{B}$ on $P_{i,i+3}$ and let $\mathcal{Q} \supseteq \{A, B\}$ be a cover for $\mathcal{B}$. The bag $\mathcal{B}$ contains the intersection of two bags that are consecutive in the list $\mathcal{B}_i, \mathcal{B}_{i+1}, \mathcal{B}_{i+1}, \mathcal{B}_{i+3}$. Therefore, at least one of $x_{i+1}, x_{i+2}, x_{i+3}$ is in $\mathcal{B}$. We also have that $c, d \in \mathcal{B}$ since $c, d \in \mathcal{B}_i \cap \mathcal{B}_{i+3}$. However, no hyperedge contains x_{i+1}, c, d or x_{i+2}, c, d or x_{i+3}, c, d. Thus, $|\mathcal{Q}| \ge 4$, and therefore $\mathcal{Q}$ is not part of a good cover. □

Claim 5. For every $i \in \{3, 9, 15, \cdots, 6n-3\}$, no bag on $P_{i,i+3}$ is covered by a set of hyperedges $\mathcal{Q}$ with $C, D \in \mathcal{Q}$ in a good cover.

Proof. The proof is symmetric to the proof of Claim 4. □

Consider a good cover of $\mathcal{D}$. A *switch* is an inclusion-wise minimal subpath $(Y_i, \ldots, Y_j)$ of the backbone of $\mathcal{D}$ where Y_i is covered by $\mathcal{Q}_i$ with $C, D \in \mathcal{Q}_i$ and Y_j is covered by $\mathcal{Q}_j$ with $A, B \in \mathcal{Q}_j$. The *signature* of a good cover of $\mathcal{D}$ is its number of switches.

Claim 6. Each good cover of each good tree decomposition of H_n has signature n.

Proof. The claim follows from Claims 3, 4, and 5. □

Due to Claim 6, we can speak of the signature of H_n and the signature of a good tree decomposition of H_n as the signature of some good cover of such a tree decomposition.

Let $H = H_n$ and $H' = H_m$. Consider a tree decomposition $\mathcal{D} = (\{X_i : i \in I\}, T)$ of $H \oplus_\mathrm{h} H'$ with generalized hypertree width 4. Without loss of generality, suppose the bags $\mathcal{B}_{-1} = S \cup \{c, d, y, z\}$ and $\mathcal{B}_{6n+1} = T \cup \{a, b, y, z\}$ are leafs of this decomposition and their neighboring bags contain both copies of x_1 and x_{6n}, respectively. Let $\mathcal{D}_{|_H}$ denote the restriction of $\mathcal{D}$ to H, i.e., it has the same tree, but each bag is restricted to the vertices of H.

Claim 7. $\mathcal{D}_{|_H}$ is a good tree decomposition for H.

Proof. Consider a bag $\mathcal{B}$ of $\mathcal{D}$ besides $\mathcal{B}_{-1}$ and $\mathcal{B}_{6n+1}$. The bag $\mathcal{B}$ contains a copy of some x_i from H'. This vertex is covered by some hyperedge from H' that does not belong to the boundary. Therefore, $\mathcal{B}_{|_H}$ is covered by at most 3 hyperedges. Suppose $\mathcal{B}_{|_H}$ is covered by exactly 3 hyperedges. Then, the cover of $\mathcal{B}_{|_{H'}}$ contains at most one hyperedge that does not belong to the boundary. But, since each such hyperedge covers at most 2 vertices among $\{a, b, c, d\}$, the cover of $\mathcal{B}$ contains at least 2 boundary hyperedges. This proves the claim. □

Symmetrically, $\mathcal{D}_{|_{H'}}$ is a good tree decomposition for H'. Since $\mathcal{D}_{|_H}$ and $\mathcal{D}_{|_{H'}}$ have the same signature, we conclude that $n = m$ due to Claim 6. This proves that the canonical right congruence $\sim_{\mathcal{H}(4\text{-GHTW})}$ does not have finite index over $\mathcal{H}(\mathcal{U}_{t,2}^{\text{small}})$.

Let k-HTW be the family of all incidence graphs G such that $\mathcal{H}(G)$ has hypertree width at most k. Let k-FHTW be the family of all incidence graphs G such that $\mathcal{H}(G)$ has fractional hypertree width at most k. To see that the proof of Theorem 4 applies to $\sim_{\mathcal{H}(4\text{-HTW})}$, observe that in our construction every hyperedge covering a bag is a subset of that bag. To see that it extends to $\sim_{\mathcal{H}(4\text{-FHTW})}$, observe that for every bag $\mathcal{B}_i$, $0 \leq i \leq 6n$, an optimal fractional cover is integral, and Claim 4 can be extended to $A, B \in \mathcal{Q}$ with weight 1—similarly for Claim 5. This completes the proof of Theorem 4.

C.2 Arguments in favour of Conjecture 1

The number of equivalence classes observed in the proof of Theorem 4 entails a lower bound on the amount of information that needs to be maintained by an algorithm when it decides whether a given hypergraph has (generalized, fractional) hypertree width k using a tree decomposition of the incidence graph: typical such algorithms implicitly remember for each bag of the tree decomposition which equivalence classes of the bag under consideration can complete into a graph with generalized hypertree width k.

However, restricting the construction in the proof of Theorem 4 to at most n vertices, the proof of Theorem 4 exhibits a class $\mathcal{C}$ of t-boundaried hypergraphs on at most n vertices with constant incidence treewidth and constant t for which the canonical right congruence has $\Omega(n)$ equivalence classes. Now, consider a class $\mathcal{C}'$ of $O(k)$-boundaried hypergraphs where each hypergraph contains k copies of hypergraphs from $\mathcal{C}$ and has at most n' vertices. Then, the number of equivalence classes of the canonical right congruence is $\Omega((n'/k)^k)$ for $\mathcal{C}'$. Hence, we conjecture that an algorithm with running time $f(k) \cdot n^c$ for a constant c and a computable function f does not exist.