Myhill-Nerode Methods for Hypergraphs

René van Bevern[1]

joint work with

Michael R. Fellows[2], Serge Gaspers[3], Frances Rosamond[2]

[1]Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany

[2]School of Engineering and IT, Charles Darwin University, Darwin, Australia

[3]University of New South Wales and National ICT Australia, Sydney, Australia

ISAAC 2013, Hong Kong

Solving Problems on Graphs of Constant Treewidth

Common ways to solve graph problems (e. g. k-Vertex Cover) in linear time on graphs of constant treewidth:

- express the problem in a formula of monadic second order logic (MSO): **Courcelle's Theorem**
- use dynamic programming over a tree decomposition

Herein, *treewidth* measures the “tree-likeness” of a graph.

No success with these methods? Try proving that the problem is W[1]-hard, that is, presumably not solvable in $f(t)\operatorname{poly}(n)$ time on n-vertex graphs of treewidth t at all.

Application Fields of Myhill-Nerode Analogs

No success with monadic second order logic (MSO), dynamic programming, and W[1]-hardness?

Then solve the problem in linear time on graphs of bounded treewidth using a different technique: **Myhill-Nerode**,

or, using **Myhill-Nerode**, show that the problem is not expressible as MSO formula,

and that the problem is not solvable using “simple” dynamic programming algorithms.

Myhill-Nerode Theorem for Regular Languages

Definition. For a language $L \subseteq \Sigma^*$, define an *equivalence relation* $\equiv_L$ over Σ^* by:

$$u \equiv_L v :\iff \forall w \in \Sigma^* : uw \in L \iff vw \in L$$

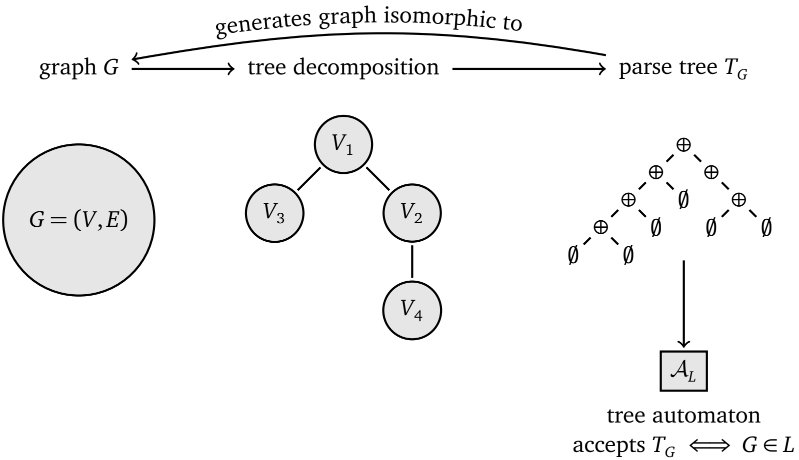
Example. Let $L = \{a^m b^n \mid m, n \in \mathbb{N}\}$, then $a \equiv_L aa$ but $a \not\equiv_L ab$

Theorem. *For a language $L \subseteq \Sigma^*$, the following are equivalent:*

- *L is regular (i.e. $x \in L$ is decidable by a finite state machine)*
- *$\equiv_L$ has a finite number of equivalence classes (finite index)*

Recognizing Graphs by Finite State Machines

Instead of creating words by concatenating other words, create graphs from other graphs using some set of operators $\oplus$, $\emptyset$, ...



Myhill-Nerode Theorem for Graphs

For which L there is a tree automaton $\mathcal{A}_L$ that accepts T_G iff $G \in L$?

Sufficient: MSO-expression for "$G \in L$" *[Courcelle's Theorem]*
Courcelle conjectures necessity; proven for special graph classes

Sufficient and necessary conditions: Myhill-Nerode for graphs, used to show that *[Abrahamson, Fellows, 1993]*

- it is linear-time decidable whether a graph has cutwidth k for any constant k
- the graph property of having constant bandwidth b is not expressible in monadic second order logic

It was later shown that checking whether a graph has bandwidth at most b is W[1]-hard *[Bodlaender, Fellows, Hallett, 1994]*

Myhill-Nerode for more than Graphs

We generalize the Myhill-Nerode theorem for graphs to

- graphs with vertex annotations
- hypergraphs

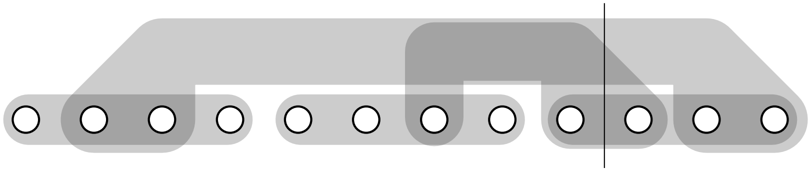
We apply the Myhill-Nerode theorem for hypergraphs to

- Hypergraph Cutwidth
- (Generalized, Fractional) Hypertree Width

Hypergraphs and Hypergraph Cutwidth

A *hypergraph* is a pair $G = (V, E)$, where

- V are *vertices*,
- $E \subseteq 2^V$ are *edges*.



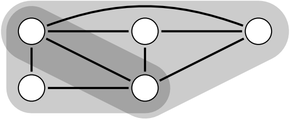
Hypergraph Cutwidth.
A *linear layout* is a map $f : V \to \mathbb{N}$.

The *width* of f is $\max_{i \in \mathbb{R}} |\{e \in E \mid \exists u, v \in e, f(u) < i < f(v)\}|$.

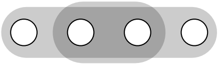
The *cutwidth* of G is the minimum width over all linear layouts.

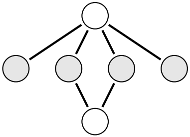
Other Width Measures for Hypergraphs

Treewidth: minimal possible width of a tree decomposition of its Gaifman graph



Incidence Treewidth: Treewidth of its incidence graph





Generalized Hypertree Width: defined using tree decompositions of the Gaifman graph, but their width is measured differently

$$\text{treewidth} + 1 \geq \text{incidence treewidth}$$

$$\text{incidence treewidth} + 1 \geq \text{generalized hypertree width}$$

Known and Our Results

Sufficient and necessary conditions: when can a tree automaton decide a hypergraph property using a tree decomposition of its incidence graph?

Does a hypergraph have cutwidth at most k?

- NP-hard even for ordinary graphs *[Gavril, 1977]*
- decidable in $O(n^{k^2+3k+3})$ time, or *[Miller, Sudborough, 1991]*
- linear-time for constant k on graphs *[Abrahamson, Fellows, 1993]*

New: linear-time for constant k on hypergraphs

Does a hypergraph have generalized hypertree width at most k?

- NP-hard already for $k = 3$ *[Gottlob, Miklós, Schwentick, 2009]*

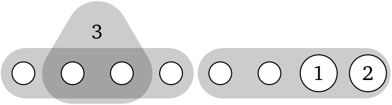
New: not decidable by a tree automaton

Proof leads us to conjecture W[1]-hardness with respect to the parameter "incidence treewidth".

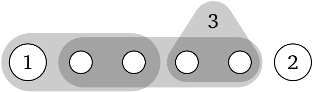
Myhill-Nerode for Hypergraphs: Gluing

Need a concept corresponding to the concatenation of words

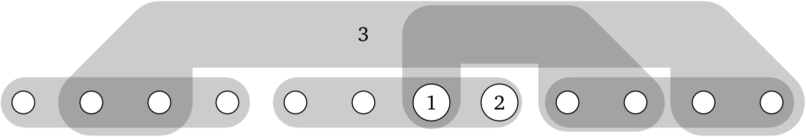
⇝ *gluing* of hypergraphs.



A 3-boundaried hypergraph G_1.



A 3-boundaried hypergraph G_2.



The glued 3-boundaried hypergraph $G_1 \oplus G_2$.

Myhill-Nerode for Hypergraphs

We show that any graph of incidence treewidth $t-1$ can be generated by $\oplus$ and some basic operators on t-boundaried graphs.

Definition. Let $\mathcal{H}_t$ be the universe of all t-boundaried hypergraphs and $F \subseteq \mathcal{H}_t$. We define an equivalence relation $\equiv_F^t$ over $\mathcal{H}_t$:

$$G_1 \equiv_F^t G_2 \iff \forall H \in \mathcal{H}_t : G_1 \oplus H \in F \iff G_2 \oplus H \in F$$

Theorem. *Let $F \subseteq \mathcal{H}_t$ be such that the hypergraphs in F have incidence treewidth at most t. The following are equivalent:*

- *$\equiv_F^t$ has a finite number of equivalence classes.*
- *F is recognizable by a finite tree automaton.*

A Tractability Result: Solving Hypergraph Cutwidth

We consider the set k-HGCW := $\{G \in \mathcal{H}_t \mid G$ has cutwidth $k\}$,

prove that graphs in k-HGCW have incidence treewidth k, and

prove that $\equiv^t_{k\text{-HGCW}}$ has finite index.

Tree automaton magic does the rest.

Proofs may disregard computability or efficiency.

How to Show Finite Index for Hypergraph Cutwidth?

Similarly as for Cutwidth on ordinary graphs, but more technical.

[Abrahamson, Fellows, 1993]

1. Define some set $\mathcal{T}$ of "tests" that a hypergraph can "pass" and let $G_1 \equiv_{\mathcal{T}} G_2$ iff they satisfy the same subset $\mathcal{T}' \subseteq \mathcal{T}$ of tests.
2. Show that $G_1 \equiv_{\mathcal{T}} G_2$ implies $G_1 \equiv^t_{k\text{-HGCW}} G_2$.
3. Show that for each test $T \in \mathcal{T}$, a hypergraph G passes T if and only if it passes $T' \in \mathcal{T}$ for some T' with $|T'| \in O(1)$.

We get: constant number of relevant tests

- $\rightsquigarrow$ constant number of possible subsets that can be passed
- $\rightsquigarrow$ constant number of equivalence classes

An Intractability Result: Hypertree Width

We construct a family G_n of t-boundaried hypergraphs with

- n vertices and incidence treewidth at most t, and
- $G_n \oplus G_m$ has generalized hypertree width 4 $\iff m = n$.

Implication: $\Omega(n)$ equivalence classes of $O(n)$-vertex hypergraphs.

Testing for constant generalized hypertree width k on hypergraphs with constant incidence width t:

- consider family of k disjoint copies of G_i for various $i \in O(n)$,
- it contains $k(t+1)$-boundaried graphs of at most $O(n)$ vertices
- and partitions into $\Omega(n^k)$ equivalence classes.

Conjecture: Generalized hypertree width is W[1]-hard with respect to the parameter incidence treewidth.

Conclusion

We have seen

- hypergraph cutwidth $\leq k$? Linear-time solvable for fixed k:
- generalized hypertree width $\leq k$? Not decidable by tree automaton for constant incidence treewidth.

Open Questions

- Solve not only the decision but also the search problem.

  [Thilikos, Serna, Bodlaender, 2005, for cutwidth]
- Is Generalized Hypertree Width W[1]-hard with respect to the parameter incidence treewidth?