Myhill-Nerode Methods for Hypergraphs

René van Bevern\textsuperscript{1}, Michael R. Fellows\textsuperscript{2},
Serge Gaspers\textsuperscript{3}, and Frances A. Rosamond\textsuperscript{2}

\textsuperscript{1} Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany
rene.vanbevern@tu-berlin.de
\textsuperscript{2} School of Engineering and IT, Charles Darwin University, Darwin, Australia
\{michael.fellows,frances.rosamond\}@cdu.edu.au
\textsuperscript{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 \cite{8} 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 \cite{7}, 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 \cite{14} and the treewidth of the dual graph \cite{18}.

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 \cite{11} and fixed-parameter tractable \cite{2} Graph Cutwidth problem, for which several fixed-parameter algorithms are known.
Cahoon and Sahni \[5\] 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 \[16\] designed an algorithm with running time \(O(n^{k^2+3k+3})\). We suspect that the framework of Nagamochi \[17\] 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 \[16\] and references therein.

**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 \[12, 13, 15\]. 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 a full version of this article \[2\].

**Preliminaries.** We use the standard graph-theoretic notions of Diestel \[6\].

**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 \(G(H)\), 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 \(I(H)\), 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 \(v \in e\).