Annals of Pure and Applied Logic 167 (2016) 1093–1122

Contents lists available at ScienceDirect

Annals of Pure and Applied Logic

www.elsevier.com/locate/apal

On (uniform) hierarchical decompositions of finite structures and model-theoretic geometry

Cameron Donnay Hill

A R T I C L E I N F O

Article history:
Received 6 May 2014
Received in revised form 5 April 2016
Accepted 12 April 2016
Available online 26 April 2016

MSC:
03C45
03C13
03D15
03B70

Keywords:
Rosy theories
Finite structures
Fraisse classes
Coordinatization
Hierarchical decomposition
Algorithmic meta-theorems

A B S T R A C T

Using finite directed systems defined from "primitive" extension and amalgamation operations, we define an abstract notion of hierarchical decomposition that applies to a large family of classes of finite structures (tame classes). We prove that for any such class **C** that is *uniformly hierarchical* – in the sense that cofinally-many members of **C** have decompositions according to a functorial "program" – the theory $T_{\mathbf{C}}$ of the generic structure is rosy. Conversely, we also show that for any tame class **C**, if $T_{\mathbf{C}}$ is rosy, then **C** is uniformly hierarchical. Thus, the project of stratifying the complexity of computationally hard problems through parametrizing "width" notions – an important current in Finite Model Theory and Descriptive Complexity Theory – has a second face in Geometric Model Theory.

0. Introduction

Using finite directed systems defined from "primitive" extension and amalgamation operations, we define an abstract notion of hierarchical decomposition that applies to a large family of classes **C** of finite structures (tame classes, such as the class of all finite graphs). This notion of hierarchical decomposition generalizes, for graphs, the well-known scheme of tree-decomposition and such slightly less well-known schemes as rank-width (see [14] and [20], for example), and for application to more general structures than graphs, our notion of decomposition does not require imposition of or reduction to a graph such as the Gaifman graph of a relational structure.

We prove that for any *tame*[1] class **C** that is *uniformly hierarchical* – in the sense, that cofinally-many members of **C** have decompositions according to a functorial program – the theory $T_{\mathbf{C}}$ of the generic/limit

E-mail address: cdhill@wesleyan.edu.

[1] Compared to [10], the definition of "tame" used here is a bit stronger: We require that $T_{\mathbf{C}}$ has weak elimination of imaginaries and a certain "bounded algebraic arity" property.

http://dx.doi.org/10.1016/j.apal.2016.04.015

structure is rosy (equivalently, here: super-rosy of finite $U^{þ}$-rank). *þ-Independence* – the defining independence relation of rosiness – is at the outer extreme of useful concepts of model-theoretic geometry (simultaneously generalizing linear independence in vector spaces, algebraic independence in fields, and topological dimension in the context of analytic and real algebraic geometry). What's more, it can be shown (see [8,2,11]) that a theory T may admit a model-theoretic independence relation *at all* if and only if T is rosy, in which case every independence relation is a refinement of þ-independence. Our proof that every uniformly hierarchical class $\mathbf{C}$ is rosy proceeds by extracting an independence relation directly from decompositions, and it is hoped that this technique could be used more broadly in the analysis of algorithms on structures in the sense of [1].

Conversely – using relatively classical techniques of coordinatization from model theory – we also show that for any tame class $\mathbf{C}$, if $T_{\mathbf{C}}$ is rosy, then $\mathbf{C}$ is uniformly hierarchical. Thus, the project of stratifying the complexity of computationally hard problems through parametrized "width" notions – an important current in Finite Model Theory and Descriptive Complexity Theory – has a second face in Geometric Model Theory. In summary, the contribution of this article is precisely the following:

Theorem 0.1. *A tame class* $\mathbf{C}$ *in a finite language is rosy if and only if it is uniformly hierarchical.*

0.1. Hierarchical decompositions in fixed-parameter tractability

The concepts of tree-decompositions and tree-width of undirected graphs formed an essential component of the proof of the celebrated Graph Minor Theorem of Robertson and Seymour (surveyed in [17] and [14]), and on graphs of bounded tree-width, many otherwise *hard* problems are known to admit efficient algorithms. A well-known example of this phenomenon is Courcelle's Theorem [6], asserting that on any class of graphs of bounded tree-width, each graph property expressible in monadic second-order logic (MSO) is decidable in *linear* time. Among the MSO-expressible graph properties are some classic `NPtime`-complete problems such as 3-Colorability, but the power of Courcelle's Theorem is already apparent on the level first-order expressible graph properties, as we briefly discuss now. Consider the following FO-Model-checking problem:

> *Given a finite graph $G = (G, R^G)$ and a first-order sentence ϕ in the language of graphs, return* True *just in case $G \models \phi$. (Otherwise, return* False*.)*

The naïve approach to this problem using just the definition of satisfaction for first-order formulas yields an algorithm with running-time $O(|G|^{\mathrm{qr}(\phi)})$ where $\mathrm{qr}(\phi)$ denotes the quantifier-rank of ϕ. There are, of course, *two parameters* involved in the expression of the running-time, but in the naïve algorithm, it is not at all clear how these might be extricated from one another. Courcelle's approach (and similar approaches for other notions of decomposition) splits the complexity (=running-time) into its two natural components – yielding algorithms with running-times expressible in the form $O(f(\mathrm{qr}(\phi)) + |G|^k)$, where k is an integer and $f : \omega \to \omega$ is a computable function. This split reflects a split in the underlying algorithms, which commonly have the following form (given G and ϕ as above):

(1) Preprocess ϕ (in time $f(\mathrm{qr}(\phi))$) into the appropriate form.
(In the case of Courcelle's algorithm, ϕ is converted into a tree automaton. For some other notions of decomposition, prenex normal form with negations only on literals is sufficient.)
(2) In polynomial time, compute a hierarchical decomposition of G (a tree-decomposition of width less than the given bound).
(3) In polynomial time, evaluate "$G \models \phi$?" (For Courcelle's algorithm, one more or less just runs the automaton on the tree just recovered; in general, a dynamic programming approach is often adequate.)