

Research Résumé

I. Finitely Presented Structures

In algorithmic model theory one studies algorithmic properties of infinite structures, that is, one considers finite representation of structures and operations on such representations. The two central questions concerning a given encoding of structures are:

- (1) Which structures admit a representation of this kind?
- (2) Which operations on these representations are computable?

Given a certain kind of representation it is usually straightforward to deduce its algorithmic properties. But the question of which structures one can encode in this way is frequently highly nontrivial.

(a) Many of the classes considered in the literature that originally were defined via automata or term rewriting systems can be characterised in logical terms. In [T3,C5,C2,J7] we investigate the class of automatic structures. In particular, it is shown that these structures are exactly those that are first-order interpretable in a certain base structure.

(b) Similar characterisations based on monadic second-order interpretations exist for tree-interpretable structures and the structures in the Caucal hierarchy. The papers [T2,J5] contain an algebraic characterisation of all structures that are monadic second-order interpretable in some tree. In [C3,J8] we show that every tree-interpretable structure can be axiomatised in guarded second-order logic.

(c) A large class of structures with a decidable monadic second-order theory is the Caucal hierarchy. The first level of this hierarchy consists of the tree-interpretable structures. In [J3] we study methods to prove that certain structures do not belong to a given level of the hierarchy.

II. Model Constructions and Decompositions

A useful tool in algorithmic model theory are operations that are compatible with a given logic in the sense that, when applying such an operation to given structures one can compute the theory of the resulting structure from the theories of the original ones. Classical examples of such operations are Feferman-Vaught like products (for first-order logic) and the generalised sums of Shelah (for monadic second-order logic).

These operations have many applications. First of all they serve as reduction methods between classes of structures. For instance, any structure in which we can interpret some structure with an undecidable theory has itself a theory that is undecidable.

Secondly, we can use compatible operations to define classes of finitely presented structures with given properties. Fix a countable set B of basic structures with the desired properties and a countable set O of operations that preserve them. Let C be the class of all structures that can be obtained from those in B by applications of the operations in O . By construction, every structure in C has the desired properties. Furthermore, each of them can be represented by a finite term over $O \cup B$. There exists a natural way to decompose the structures in C since each of them is build up from the simple structures of B . Depending on the particular choice of operations this can lead to the development of a structure theory for C .

A prominent example of this approach is the class of structures of bounded partition width which can be characterised in the following equivalent terms:

- (1) as having a hierarchical decomposition where a certain complexity measure is bounded;
- (2) as being monadic second-order interpretable in a tree;
- (3) as being build up from finite structures with the help of (i) disjoint unions and (ii) quantifier-free interpretations.

(a) The survey [H1] gives an overview of this approach with an emphasis on operations that are compatible with either first-order logic or monadic second-order logic.

(b) The article [J4] studies various algebras of finite structures and the corresponding notions of recognisable and equational classes. The former are those saturated by a congruence of finite index, the latter are the solutions to finite systems of equations.

(c) [H2] is a survey on an iteration operation, introduced by Muchnik, that is compatible with monadic second-order logic. The article [J6] extends these results by showing that the Muchnik iteration is also compatible with several extensions of monadic second-order logic.

(d) In [J5,T2] we study the class of structures that can be interpreted in a tree. Several equivalent characterisations are presented and it is shown that one can use tools from first-order model theory to study such structures.

III. Model Theory for Monadic Second-Order Logic

Within the last two decades the beginnings of a model theory for monadic second-order logic have emerged. On the one hand there are structures whose monadic theory is simple enough to develop a structure theory. So far, all known examples of such structures can be interpreted in some tree.

(a) The theory of such structures is the topic of [J5,T2]. These two papers introduce the notion of partition width, a generalisation of clique width. We study classes of structures of bounded partition width and we give several characterisations of such classes.

(b) A large class of structures with a decidable monadic second-order theory is the Caucal hierarchy. Each of these structures has a finite partition width. In [J3] we study methods to prove that certain structures do not belong to a given level of the hierarchy.

(c) On the other hand there are the structures where the monadic theory is extremely complicated. Prominent examples are structures containing large definable grids. According to a conjecture of Seese these two extremes form a dichotomy: either a structure can be interpreted in a tree or it contains a large grid. Partial results concerning this conjecture appear in [U3,U4].

(d) If, instead of monadic second-order logic, one considers a logic where one can quantify both over sets of vertices and sets of edges one obtains guarded second-order logic. Structures with a manageable guarded second-order theory are necessarily sparse, i.e., they have few edges. According to a result of Courcelle, on countable sparse structures the expressive power of guarded second-order logic collapses to that of monadic second-order logic. In [J1] we study the expressive power of guarded second-order logic. In particular, we extend the result of Courcelle to sparse structures of arbitrary cardinality.

(e) We can use interpretations to reduce one class of structures to another one. For classes of finite structures a complete description of the resulting hierarchy is presented in [J2].

IV. Automata Theory and Algebraic Language Theory

There is a well-established connection between automata and the monadic second-order theories of certain structures, like the order of the natural numbers or the infinite binary tree. In particular, one obtains decision procedures for these theories by translating formulae into automata and then checking their emptiness.

Instead of using monadic second-order logic or automata, one can also characterise recognisable languages also by algebraic means via homomorphisms into finite algebras. This connection has recently been extended to other structures, such as infinite words and trees.

(a) Both, monadic second-order logic and the algebraic approach can be used to generalise formal language theory to arbitrary structures. For certain kinds of graph algebras this was done in [J4]. This paper extends formal language theory to various algebras of finite structures. We study the corresponding algebraic notions of recognisable and equational classes, and we relate them to the notion of definability in monadic second-order logic.

(b) Besides considering automata as recognisers of languages, we can also use them to present infinite structures. For instance, the graphs in the Caucal hierarchy coincide with the configuration graphs of higher-order pushdown automata. In [J3] we use higher-order pushdown automata to study the classes in the Caucal hierarchy. In particular, we develop methods to prove that certain structures do not belong to a given level of the hierarchy. The main technical result is a pumping lemma for these automata.

(c) Recently, the algebraic approach to formal language theory has been extended to languages of infinite words and of finite trees. For infinite trees there is no such theory since the required combinatorial techniques have not been developed yet. The paper [U2] makes a first contribution in this direction.

V. Finite model theory, descriptive complexity theory, and algorithmic issues

For many applications, one needs logics with the right balance of expressive power and algorithmic manageability. In many cases, in particular in verification and database theory, one can obtain such logics by extending some weak logic by fixed-point operators. This has led to a wide range of fixed-point logics.

(a) [H3] is a survey on guarded fixed-point logic. We present automata-based algorithms for model checking and satisfiability testing for this logic, and we study the complexity of these problems.

(b) In [C1] we study fixed-point inductions of a monadic second-order formulae on finite words. The main result is a proof that it is decidable whether the length of these inductions is uniformly bounded.

(c) Descriptive complexity theory studies the correspondence between the computational complexity of classes of finite structures and the logics these classes can be axiomatised in. In [C4] we introduce a different setting where we consider the complexity and definability of sets definable in a fixed structure. Several complexity classes are characterised in this way.

(d) Ehrenfeucht-Fraïssé Games are one of the main model-theoretic tools in finite model theory. Unfortunately, on nontrivial structures the combinatorics involved in playing these games quickly become unmanageable. In [U5] we study several ways to simplify games and to decompose them into simpler subgames. While in the literature one mostly considers games on sparse structures, in this article we place the emphasis on structures that are not sparse.