Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research
Report
RR-95-05

A First-Order Axiomatization of the Theory of Finite Trees

Rolf Backofen, James Rogers, K. Vijay-Shanker

May 1995

Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH

Postfach 20 80
D-67608 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-66123 Saarbrücken, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern and Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Atlas Elektronik, Daimler-Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Mannesmann-Kienzle, Sema Group, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct systems with technical knowledge and common sense which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

□ Intelligent Engineering Systems
□ Intelligent User Interfaces
□ Computer Linguistics
□ Programming Systems
□ Deduction and Multiagent Systems
□ Document Analysis and Office Automation.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Dr. Dr. D. Ruland
Director

A First-Order Axiomatization of the Theory of Finite Trees

Rolf Backofen, James Rogers, K. Vijay-Shanker

DFKI-RR-95-05

To appear in Journal of Logic, Language and Information. This report is also available from the Institute for Research in Cognitive Science, Univ. of Pennsylvania, as report number 95-02, and from the Univ. of Delaware as report number NLP-HCI-AE-95-01

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITWM-ITW 01 IV 101 K/1 (VERBMOBIL)).

ISSN 0946-008X

A First-Order Axiomatization of the Theory of Finite Trees

Rolf Backofen [*]
DFKI Saarbrücken
Stuhlsatzenhausweg 3
66123 Saarbrücken, Germany

James Rogers
Institute for Research in Cognitive Science
Univ. of Pennsylvania
Philadelphia, PA 19104-6228, USA

and

K. Vijay-Shanker [†]
Dept. of Computer and Information Science
Univ. of Delaware
Newark, DE 19716, USA

Abstract. We provide first-order axioms for the theories of finite trees with bounded branching and finite trees with arbitrary (finite) branching. The signature is chosen to express, in a natural way, those properties of trees most relevant to linguistic theories. These axioms provide a foundation for results in linguistics that are based on reasoning formally about such properties. We include some observations on the expressive power of these theories relative to traditional language complexity classes.

Key words: Trees, First-Order Theories, Axiomatizations, Natural Language Syntax, Ehrenfeucht-Fraïssé Games

1. INTRODUCTION

There has been, over the last ten or fifteen years, a growing body of research in generative and computational linguistics that depends to a great extent on reasoning formally about trees. For example, there are a number of grammatical formalisms that have been proposed that manipulate logical descriptions of the trees representing the syntactic structure of strings rather than strings or the trees themselves (Marcus *et al.*, 1983; Henderson, 1990; Vijay-Shanker, 1992). Parsing, in these formalisms, is a process of constructing a formula that characterizes the trees that yield a given input. Recognition is the question of whether that formula is satisfiable. These formalisms, then, presuppose a means of manipulating these formulae and determining their satisfiability. In other works a logical language is used to formalize the grammatical framework itself (Johnson, 1989; Stabler, Jr.,

[*] The research reported in this paper has been supported by the Bundesminister für Forschung und Technologie under contract ITW 01 IV 101 K/1 (Verbmobil).

[†] Part of this work was done during the period when this author was at DFKI, Saarbrücken. This author would like to thank Prof. Dr. H. Uszkoreit and Prof. Dr. W. Wahlster for providing the facilities and a supportive environment.

1992; Blackburn *et al.*, 1993). The intent here is to translate a given grammar G into a formula ϕ_G such that the set of trees generated by the grammar is exactly the set of trees that satisfy ϕ_G. Parsing, then, is just identifying the set of models of ϕ_G that yield a given string. Recognition can be understood as the problem of determining if a formula asserting that the yield of a tree is a given string is consistent with ϕ_G. Such an approach can provide the foundation for a formal approach to issues about the grammar formalism itself. Thus formalizations of this sort have formed the basis of arguments about the consistency and independence of various sets of principles (Stabler, Jr., 1992), of accounts of certain linguistic phenomena (Cornell, 1992), and of results relating to the fundamental properties of linguistic structures (Kayne, 1994; Kracht, 1993). The readers of this volume will likely be familiar with many other examples as well.

The goal of the work reported here is to provide a key portion of the foundation of such arguments—a set of first-order axioms from which all of the first-order properties of finite trees can be derived.

There have been two dominant approaches to the formalization of trees. One of these, an algebraic approach, has grown primarily from studies in the semantics of programming languages and program schemes (Courcelle, 1983). In this approach, trees interpret terms in the algebra generated by some finite set of function symbols. The term $f(x, y)$, for instance, is interpreted as a tree in which the root is labeled f and has the subtrees x and y as children. Maher (1988) has provided an axiomatization for the equational theory of these trees. For our purposes, the characteristics of this theory which are most significant are its domain—in it one reasons about (i.e., variables range over) entire trees as opposed to individual nodes in those trees—and the fact that equality in the theory is extensional in the sense that $f(x, y) = f(g(a), g(a))$ implies that $x = y$.

In contrast, the second approach is concerned with the internal structure of trees. Formal treatments of trees of this sort are ultimately founded in the theory of multiple successor functions, a generalization of the theory of the natural numbers with successor and less-than. The domain of this theory is the individual nodes in the tree—one reasons about the relationships between these nodes. Here, it is a theorem that the left successor of a node is not equal to the right successor of that node regardless of how the nodes are labeled. The structure of multiple successor functions is an infinite tree in which all nodes have the same (possibly infinite) degree. Its language includes symbols for each successor function, a symbol for less-than, and one for lexicographic order (the total order imposed by less-than and the ordering of the successor functions). Rabin (1969) has shown that *SnS*, the monadic second-order theory of this structure, is decidable. An axiomatization of the weak monadic second-order fragment has been provided by Siefkes (1978). The set-theoretic component of this axiomatization is crucial to its completeness.

In applications to linguistics, trees typically represent the relationships between the components of sentences. Here, it is the second approach that is appropriate. One wants to distinguish, for instance, between identical noun phrases occurring at different positions in a sentence. These applications are concerned with finite trees with variable branching. The relations of interest are based on the relation of a node to its immediate successors (parent or immediate domination), the relation of a node to the nodes it is less-than, i.e., nodes in the subtree rooted at that node (domination), and the left-to-right ordering of the branches in the tree (precedence or left-of). Here, as in SnS, it is often useful (as in Marcus *et al.* (1983), Henderson (1990), Cornell (1992), Vijay-Shanker (1992), and Rogers and Vijay-Shanker (1994), for example) to be able to reason about domination independently of parent. Unlike SnS, though, it is also often useful to reason about the parent relation independently of left-of.

We will focus on two classes of finite trees. In the first of these the number of children of any node is bounded by a constant. The existence of such a bound is typical of the trees derived in a number of grammar formalisms, including Context-Free and Tree-Adjoining grammars, and is a principle of some linguistic theories (Kayne, 1981). We refer to this as the class of finite trees with *bounded branching.* In the second class, nodes may have any finite number of children. Such trees arise in certain accounts of coordination and when grammar formalisms allow the use of regular expressions in rewriting rules (as in Generalized Phrase Structure Grammar (Gazdar *et al.*, 1985)). We say such trees are finitely branching. The class of such trees, of course, includes the trees with bounded branching, and we refer to this larger class simply as the class of finite trees. In this paper we provide first-order axiomatizations of the theories of these two classes of trees in a signature including the parent, domination, and left-of relations. This signature is comparable to those that have been employed in most of the linguistic works on the formal properties of trees. Thus the language of these theories is tailored to the range of applications that are our primary interest. Further, as they are purely first-order axiomatizations, they provide a basis for reasoning about the elementary properties of trees without appealing (as in the Siefkes axiomatization) to the higher-order fragment of their theory.

Typically, in the literature, formal results about the properties of trees are based on partial enumerations of their fundamental properties, that is, on partial sets of axioms for trees (see, for example, Partee *et al.*, 1990). Such properties include the fact that domination is a discrete partial order with a minimum element (the root), the fact that left-of is a discrete linear order on the set of children of each node, and the fact that precedence is inherited in the sense that the nodes preceding a given node also precede all its descendants. In Section 2, we give a set of axioms $\mathcal{A}$ that capture these fundamental properties. We show, however, that these axioms do not define exactly the set of finite trees, and, in fact, that no set of first-order axioms can do so. For this reason, we focus not on axiomatizing finite trees as a class of mathematical structures, but rather on axiomatizing the theory of that class of structures—the set of properties that are true in all finite trees.

The key properties that $\mathcal{A}$ misses are the facts that induction on the depth of a node and on the number of siblings preceding a node are valid on these structures, and that every branch and every set of siblings is finite. These properties are straightforward to express in monadic second-order logic. Our approach, which was originally employed by Doets (1989), is to translate the second-order axioms for these properties into first-order schemas. In this way, in Section 2.3, we develop a schema **Fin-D** capturing the property of having finite depth, and a schema **Fin-B** capturing the property of finite branching. The first of these, when coupled with an axiom bounding the number of children of any node with a constant n (which we refer to as **BBn**), suffices to extend $\mathcal{A}$ to a set of axioms $\mathcal{A}_{\mathbf{BBn}}$ that capture the first-order theory of finite trees with bounded branching. When we extend $\mathcal{A}$ with both **Fin-D** and **Fin-B** we get a set of axioms $\mathcal{A}_{\mathbf{Fin}}$ which capture the first-order theory of finite trees. To establish these claims, of course, we must show that this translation of the second-order axioms into first-order schema does not affect their first-order consequences. The proofs of these facts are given in Sections 3 and 4. In Section 3 we lay out the essential techniques and operations on models on which the proofs are built; Section 4 contains the proofs themselves. The paper closes with some observations about the expressive power of these theories.

Our results show that the basic properties of trees as usually given are not sufficient in themselves to derive all first-order properties of trees. On the other hand, arguments about the structure of trees are rarely limited to deductions from these properties. In fact inductions of the sort we capture in our schemas are nearly characteristic of such arguments. It is generally assumed that such methods do suffice. Our work, in effect, shows that this is indeed the case.

2. LANGUAGE, AXIOMS, AND MODELS

The language is an ordinary first-order language, with neither constants nor function symbols. It includes the two place relation symbols $\triangleleft, \triangleleft^*, \prec$, which represent *parent*, *domination*, and *left-of* respectively. It should be noted that this is a finite relational language with no function symbols. A number of key results established in Section 3 are based on just these properties.

Throughout this paper we use infix notation, writing, for example, $x \triangleleft^* y$ rather than $\triangleleft^*(x, y)$. We use the symbol $\triangleleft^+$ as an abbreviation for *proper domination*, i.e., domination by a path of length greater than zero. The expression $x \triangleleft^+ y$ should be taken to be equivalent to $x \triangleleft^* y \wedge x \not\approx y$.

2.1. Basic Axioms

We begin with a set of axioms that, with a couple of notable exceptions, capture all of the properties of trees encountered in the linguistic literature (as in, for instance, the definition of a tree given by Partee *et al.* (1990)). As we will see, these axioms are satisfied by a variety of structures other than trees, which accounts for the properties they fail to capture. Those properties are not first-order definable, and we will not be able to eliminate the non-standard models of our axioms. We can, however, extend them in such a way that they imply exactly the first-order theory of finite trees. We do this in Section 2.3, after we have fixed our notion of trees and considered the structure of the non-standard models.

A1 $(\exists x)(\forall y)[x \triangleleft^* y]$,

A2 $(\forall x, y)[(x \triangleleft^* y \wedge y \triangleleft^* x) \rightarrow x \approx y]$,

A3 $(\forall x, y, z)[(x \triangleleft^* y \wedge y \triangleleft^* z) \rightarrow x \triangleleft^* z]$,

A4 $(\forall x, y)[x \triangleleft y \rightarrow (x \triangleleft^+ y \wedge (\forall z)[(x \triangleleft^* z \wedge z \triangleleft^* y) \rightarrow (z \approx x \vee z \approx y)])]$,

A5 $(\forall x, z)[z \triangleleft^+ x \rightarrow (\exists y)[y \triangleleft x]]$,

A6 $(\forall x, z)[x \triangleleft^+ z \rightarrow (\exists y)[x \triangleleft y \wedge y \triangleleft^* z]]$,

A7 $(\forall x, y)[x \prec y \leftrightarrow (\neg x \triangleleft^* y \wedge \neg y \triangleleft^* x \wedge y \not\prec x)]$,

A8 $(\forall w, x, y, z)[(x \prec y \wedge x \triangleleft^* w \wedge y \triangleleft^* z) \rightarrow w \prec z]$,

A9 $(\forall x, y, z)[(x \prec y \wedge y \prec z) \rightarrow x \prec z]$,

A10 $(\forall x)[(\exists y)[x \triangleleft y] \rightarrow (\exists y)[x \triangleleft y \wedge (\forall z)[x \triangleleft z \rightarrow z \not\prec y]]]$

A11 $(\forall x)[(\exists y)[x \prec y] \rightarrow (\exists y)[x \prec y \wedge (\forall z)[x \prec z \rightarrow z \not\prec y]]]$,

A12 $(\forall x)[(\exists y)[y \prec x] \rightarrow (\exists y)[y \prec x \wedge (\forall z)[z \prec x \rightarrow y \not\prec z]]]$.

We will denote this set of axioms by $\mathcal{A}$.

A1 asserts that every tree has a root. ***A2*** and ***A3*** require domination to be anti-symmetric and transitive. ***A4*** states that a node properly dominates its child and that there is no other node in the domination path between them. ***A5*** and ***A6*** together with ***A4*** assert that domination is a discrete partial order. ***A5*** states that a node that is not a root has a parent (an immediate predecessor) and ***A6*** states that every node that properly dominates another has a child (an immediate successor) on the path to that node. ***A7*** asserts that any two nodes are related by either domination or left-of, but no nodes are related by both. It also requires left-of to be irreflexive and, consequently, implies reflexivity of domination. ***A8*** relates left-of and domination. It requires that a left-of relation between any pair of nodes is inherited by all nodes in the subtrees dominated by those nodes. ***A9*** states that left-of is transitive. ***A10*** states that any node with children has a leftmost child. That the set of children of any node are linearly ordered by left-of is a consequence of ***A7***. ***A11*** and ***A12*** together require that this linear order is discrete.

Linear branching (the fact that each node is at the end of a unique path from root) is an example of a commonly encountered property that is not explicit in these axioms but that is implied by them. Suppose x and y both lie on a path to z. Then $x \triangleleft^* z$ and $y \triangleleft^* z$. By ***A7***, either $x \triangleleft^* y$ or $y \triangleleft^* x$ or $x \prec y$ or $y \prec x$. But $x \prec y$ implies $z \prec y$ which implies $\neg y \triangleleft^* z$, by ***A8*** and ***A7***. Similarly for $y \prec x$. Thus we have either $x \triangleleft^* y$ or $y \triangleleft^* x$, that is, both x and y must lie on the same path.

2.2. Models

Models are ordinary first-order structures interpreting the predicate constants, i.e., a tuple $A = \left\langle |A|, \mathcal{I}^A, \mathcal{D}^A, \mathcal{P}^A \right\rangle$, where:

- $|A|$ is a non-empty universe,
- $\mathcal{I}^A$, $\mathcal{D}^A$, and $\mathcal{P}^A$ are binary relations over $|A|$ (interpreting $\triangleleft$, $\triangleleft^*$, and $\prec$ respectively).

When the context makes it clear, we will simply use $\mathcal{I}$ (Immediate domination), $\mathcal{D}$ (Domination), and $\mathcal{P}$ (Precedence), rather than $\mathcal{I}^A$, $\mathcal{D}^A$, and $\mathcal{P}^A$. As our aim is to axiomatize trees, if A is a model and $a \in |A|$ then we say a is a *node* in A. Likewise, if $\langle a, b\rangle \in \mathcal{I}^A$ we say a is the *parent* of b and b is a *child* of a. If $\langle a, b\rangle \in \mathcal{D}^A$ then we say *a dominates b* and b is *dominated by a*. If $\langle a, b\rangle \in \mathcal{P}^A$ we say that a is *left-of* b. If, in addition, there exists a $c \in |A|$ such that $\langle c, a\rangle, \langle c, b\rangle \in \mathcal{I}^A$ then we say a and b are *siblings* with a a *left-sibling* of b and b a *right-sibling* of a. It follows from $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ that any model A that satisfies $\mathcal{A}$ will have an unique node dominating every other node. Such a node will be called the *root* of A and will be designated by $\mathrm{r}(A)$. Given two nodes that are related by domination, we will refer to the set of nodes falling between them with respect to domination as the *path* between them. Any maximal set of nodes that is linearly ordered by (proper) domination is a *branch*. In finite trees, the branches are just the paths from the root to the *leaves* of the tree—its maximal nodes wrt domination. Finally, the *branching factor* of a node is the cardinality of the set of its children.

2.2.1. *Intended Models*

We fix our notion of trees by adopting a standard definition based on tree-domains. A tree-domain may be thought of as a set of addresses of nodes in a tree. In this address scheme, the root has address ϵ, and if a node has address u, then its children in left to right order will have addresses $u0, u1, \cdots$.

DEFINITION 1. A *tree domain* is a non-empty set $T \subseteq \mathbb{N}^*$, ($\mathbb{N}$ is the set of natural numbers) satisfying, for all $u, v \in \mathbb{N}^*$ and $i, j \in \mathbb{N}$, the conditions:

$\boldsymbol{TD_1}$ $\quad uv \in T \Rightarrow u \in T,$ $\qquad$ $\boldsymbol{TD_2}$ $\quad ui \in T, j < i \Rightarrow uj \in T.$

Every tree domain has a natural interpretation as one of our structures, and it is easy to show that this interpretation satisfies $\mathcal{A}$.

DEFINITION 2. The *natural interpretation* of a tree domain T is the structure $T^\natural = \left\langle T, \mathcal{I}^{T^\natural}, \mathcal{D}^{T^\natural}, \mathcal{P}^{T^\natural} \right\rangle$, where:

$$\begin{aligned}
\mathcal{I}^{T^\natural} &= \{\langle u, ui\rangle \in T \times T \mid u \in \mathbb{N}^*, i \in \mathbb{N}\}, \\
\mathcal{D}^{T^\natural} &= \{\langle u, uv\rangle \in T \times T \mid u, v \in \mathbb{N}^*\}, \\
\mathcal{P}^{T^\natural} &= \{\langle uiv, ujw\rangle \in T \times T \mid u, v, w \in \mathbb{N}^*, i < j \in \mathbb{N}\}.
\end{aligned}$$

LEMMA 3. *If T is a tree domain then $T^{\natural} \models \mathcal{A}$.*

Given the natural interpretation of a tree domain $T^{\natural}$ it is easy to see that for all $a \in T$ the set of nodes dominating a is finite, as is the set of left-siblings of a. That is, for any $a \in T$, the sets

$$\begin{aligned} \text{above}(a) &= \left\{ b \mid \langle b, a \rangle \in \mathcal{D}^{T^{\natural}} \right\}, \\ \text{left-sibling}(a) &= \left\{ b \mid \langle b, a \rangle \in \mathcal{P}^{T^{\natural}} \text{ and } \langle c, a \rangle, \langle c, b \rangle \in \mathcal{I}^{T^{\natural}} \text{ for some } c \in T \right\} \end{aligned}$$

are finite. The following proposition establishes that this is a sufficient condition for a structure to be isomorphic to the natural interpretation of a tree domain.

THEOREM 4. *Suppose $A = \left\langle |A|, \mathcal{I}^A, \mathcal{D}^A, \mathcal{P}^A \right\rangle$ is a model of $\mathcal{A}$ such that for all $a \in |A|$,* above(a) *and* left-sibling(a) *are finite. Then there is some tree domain T such that $T^{\natural}$ is isomorphic to A.*

Proof. Let $l_A : |A| \to \mathbb{N}^*$ be defined:

$$l_A(x) = \begin{cases} \varepsilon & \text{if } \langle y, x \rangle \notin \mathcal{I}^A \text{ for all } y \in |A| \\ l_A(y) \cdot i & \text{if } \langle y, x \rangle \in \mathcal{I}^A \text{ and} \\ & i = \mathbf{card}(\left\{ y \mid \langle y, x \rangle \in \mathcal{P}^A \text{ and } \langle z, y \rangle, \langle z, x \rangle \in \mathcal{I}^A \text{ for some } z \right\}). \end{cases}$$

Let $l(A)$ be the range of l_A. It is easy to show that l_A is total and well-defined and that $l(A)$ is a tree domain, i.e., that $l(A)$ is a non-empty subset of $\mathbb{N}^*$ that satisfies conditions $\boldsymbol{TD1}$ and $\boldsymbol{TD2}$. It follows then, from the definitions of l_A and $l(A)^{\natural}$ that A is isomorphic to $l(A)^{\natural}$.

Our intended models are isomorphic to the natural interpretations of tree domains. This gives us, of course, a class that includes both trees in which some branches may be infinite and those in which some nodes may have infinitely many children. We get the class of finite trees by requiring every branch to be finite and by restricting the number of children of any node either to be less than a fixed bound or to be finite. Henceforth, we will reserve the term "trees" for these classes of structures. The key property of these models is that all branches (ordered by proper domination) and all sets of children (ordered by left-of) are isomorphic to initial segments of the natural numbers (ordered by less-than). Thus properties of these structures can be established by induction on the depth of nodes and on the number of left-siblings. Such inductions are common (even characteristic) in arguments about the structure of trees, and the validity of induction is one of the properties of trees that is not captured by our basic axioms. The other is the fact that in finite trees all branches and all sets of siblings have a maximum node (wrt domination and left-of respectively), that is, branches and sets of siblings are isomorphic to *proper* initial segments of the natural numbers. These two properties distinguish our intended models from the non-standard models of the axioms. As they are not first-order definable properties, no set of first-order axioms will be able to eliminate the non-standard models.

2.2.2. *Non-Standard Models*

Since our intended class of structures includes trees with arbitrary finite depth and arbitrary finite branching, any first-order axiomatization will admit models in which there are paths and sets of siblings that are infinite (by compactness), and, by the upward Löwenheim-Skolem theorem, models in which these sets may have any infinite cardinality. Such non-standard models must include some node which cannot be reached by a finite path from the root or some node that has infinitely many left siblings. We will refer to such nodes as "non-standard". In this section we explore the structure of these models. We will consider first the possibility of an infinitely deep node. Note that, since all trees satisfy the axioms $\mathcal{A}$, every axiomatization of trees must imply at least these properties. By ***A*1** each such node is dominated by the root, and by linear branching it is dominated by some unique path from the root. ***A*6** ensures that each node has an immediate successor on the path to any node it properly dominates. Thus there is a sequence of nodes isomorphic to an initial segment of $\mathbb{N}$ extending from each node toward each of the nodes it dominates. This sequence forms only the initial portion of the path to a non-standard node, its *standard part*. By ***A*5**, every node other than the root has an immediate predecessor and thus there is a sequence of nodes isomorphic to $\mathbb{N}$ extending from each non-standard node toward the root. This sequence is disjoint from the standard part of the path extending toward the node from the root, of course, otherwise the node would be reachable by a finite path.

A similar analyses applies when we consider the paths from a non-standard node to the nodes it dominates. Thus the path from root to any non-standard node looks like a $\mathbb{Z}^+$-chain followed by some possibly empty sequence of $\mathbb{Z}$-chains followed by a $\mathbb{Z}^-$-chain. (where a $\mathbb{Z}^+$-chain ($\mathbb{Z}^-$-chain) is a sequence isomorphic to the positive (negative) integers when $\triangleleft^*$ is mapped to $\leq$). The overall picture, then, is a structure that includes a standard tree as a submodel, with an array of disjoint structures hanging off of its infinite branches. These structures, in turn, are "tree-like" with the exception that they have no minimum point, rather they extend infinitely down toward the root.[1] There may be any number of these non-standard segments, forming a roughly tree-like arrangement with the standard part as the root.

The case of non-standard models including points with infinitely many left-siblings is somewhat simpler. The axioms ***A*7** through ***A*12** ensure that left-of linearly orders every set of siblings, and that this ordering is discrete and has a minimum. Again an analysis similar to our discussion of the path to a non-standard node applies. Every infinite set of siblings consists of a $\mathbb{Z}^+$-chain followed

[1] These bear a relationship to $\mathbb{Z}$ that is analogous to the relationship between an infinite tree and $\mathbb{N}$.

by a (possibly empty) sequence of $\mathbb{Z}$-chains, and possibly followed by a single $\mathbb{Z}^-$ chain.

2.3. Additional Axioms

As we have just seen, the class of all and only our intended structures is not definable in first-order logic. Nonetheless, we are still able to axiomatize the *theory* of those intended structures, that is, we provide a set of axioms for which the set of first-order consequences of the axioms is exactly the first-order theory of finite trees. We already have, from Lemma 3, that every finite tree satisfies our basic set of axioms $\mathcal{A}$, thus every consequence of $\mathcal{A}$ is in the theory of finite trees. The problem is that there are properties of trees, particularly those related to the induction principle and the existence of maximum nodes, that are not true of all the non-standard models. Thus the consequences of $\mathcal{A}$ are a proper subset of the theory of finite trees. Our goal is to extend $\mathcal{A}$ with additional axioms sufficient to imply that portion of the theory that the basic axioms miss. N.B., these axioms cannot eliminate all of the non-standard models of our axioms. Rather, our additional axioms will serve to restrict those non-standard models sufficiently to guarantee that they do not affect the theory. That is, there will be no sentence that is true of all trees but false in some non-standard model of the extended axioms.

Note that the class of our intended models is definable in monadic second-order logic. If we can quantify over sets of nodes as well as individual nodes (equivalently, if we can quantify over properties of nodes) then finiteness of branches and of sets of siblings are definable properties of structures. Doets (1989) has provided a general approach to constructing first-order axiomatizations of first-order (and even universal monadic second-order) theories of monadic second-order classes of structures. The idea is to replace the second-order sentences in a monadic second-order axiomatization of the class with first-order schema. That is, replace every second-order axiom in which a term $P(x)$ occurs, where P is a variable over sets, with an infinite sequence of first-order axioms in which $P(x)$ is replaced with $\phi(x)$ for each first-order formula $\phi(x)$ (in which at most x appears free) in turn.[2] In translating the second-order axiom into a first-order schema we are, in essence, passing from quantification over arbitrary sets to quantification over *first-order definable* sets. It is not the case that such a passage will always preserve the theory. To establish that the consequences of the resulting first-order axioms are exactly the first-order consequences of the second-order axioms (i.e., the first-order theory of the intended models) we must show that every sentence that is satisfied by a model of the first-order axioms (possibly a non-standard model) is also satisfied by an intended model, i.e., a model of the second-order axioms. It will follow that every sentence that is satisfied by every standard model will also be satisfied

[2] Peano's first-order schema for induction (a monadic second-order property) is a familiar example of such a schema.

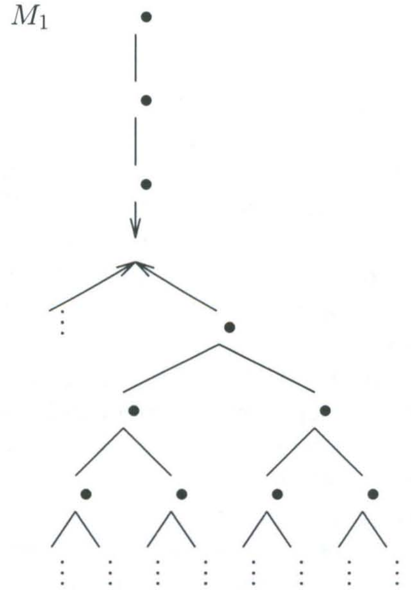

Fig. 1. A non-standard model of $\mathcal{A}$.

by every non-standard model.[3] Thus the non-standard models do not affect the theory, that is, the consequences of the axioms will coincide with the intended theory.

In the remainder of this section we follow this approach in developing schemas that, when added to our basic set of axioms $\mathcal{A}$, give us axiomatizations of the first-order theory of finite trees in which branching is bounded by a constant, and of the first-order theory of finite trees in which branching is unbounded.

2.3.1. *Finite Paths*

We will ignore, at first, the issue of infinite branching and focus on non-standard models with nodes that are infinitely deep. An example is the structure M_1 depicted pictorially in Figure 1. In this figure the solid lines represent immediate domination links, solid lines with arrows represent an infinite sequence of immediate domination links, and ellipses represent repeated structure. This model consists of a standard part in which every node has exactly one child and a single non-standard part in which every node has exactly two children. (Recall that this implies that there is an infinite sequence of nodes in the non-standard part extending towards

[3] This is because a non-standard model fails to satisfy a sentence ϕ only if it satisfies $\neg\phi$. By our result, this will necessarily be satisfied in some intended model as well. Thus ϕ can not be in the theory of the intended models.

the root from those shown in the figure, each of which has exactly one sibling.) Let binary(x) be the formula

$$(\exists x_1, x_2)[x \triangleleft x_1 \wedge x \triangleleft x_2 \wedge x_1 \prec x_2].$$

Let ψ_{wd} be the sentence

$$(\exists x)[\mathrm{binary}(x)] \rightarrow (\exists x)[\mathrm{binary}(x) \wedge (\forall y)[y \triangleleft^+ x \rightarrow \neg\mathrm{binary}(x)]].$$

This sentence asserts that if there is any node with two children then there is a minimal node (wrt domination) with two children. That this is true of all trees follows from the fact that, because all branches are isomorphic to initial segments of $\mathbb{N}$, domination in trees is a well-founded partial order. It is easy to verify that M_1 satisfies $\mathcal{A}$, but fails to satisfy ψ_{wd}. Thus ψ_{wd} is a sentence that is in the theory of finite trees but is not in the consequences of $\mathcal{A}$.

We must find an extension of $\mathcal{A}$ that implies ψ_{wd} (at least), or equivalently, that is not modeled by structures such as M_1. It is possible to restrict our models to structures in which domination is a well-founded partial order with the second-order axiom:

$$(\forall P)[\,(\exists x)[P(x)] \rightarrow (\exists x)[P(x) \wedge (\forall y)[y \triangleleft^+ x \rightarrow \neg P(x)]]\,].$$

The corresponding first-order schema is:

WF-D $(\exists x)[\phi(x)] \rightarrow (\exists x)[\phi(x) \wedge (\forall y)[y \triangleleft^+ x \rightarrow \neg\phi(x)]].$

The reader should notice that ψ_{wd} is that instance of **WF-D** in which $\phi(x)$ is the formula binary(x). Thus the addition of **WF-D** to our axioms will add ψ_{wd} to their consequences and exclude M_1 from the class of their models.

It should be noted that the class of models in which domination is a well-founded partial order is exactly the class in which induction on the depth of nodes is valid, and that the proof of this fact goes through even if we restrict ourselves to first-order definable sets. (In other words, the class of models in which induction on the depth of nodes is valid for first-order definable properties is exactly the class in which every first-order definable set has a minimum wrt domination.) Further, the class of models in which induction on the depth of nodes is valid is exactly the class of models in which every node can be reached by a finite path from the root. It remains to be shown, of course, that the theory of models in which every first-order definable set of nodes includes a minimal node coincides with the theory of models in which every set includes a minimal node.

2.3.2. *Finite Depth*

The models of **WF-D** (even in the monadic second-order form), of course, include trees with infinite branches (since it is concerned with well-foundedness, not finiteness). A standard approach to eliminating infinite branches (in monadic second-order languages) is to require every non-empty set to include a maximal node as well as a minimal node. When we are dealing with discrete partial orders, as in our case, it suffices to just require every non-empty set to have a maximal point.[4] Thus we can restrict our models to those with finite branches using the dual of the monadic second-order axiom for well-foundedness

$$(\forall P)[\,(\exists x)[P(x)] \rightarrow (\exists x)[P(x) \wedge (\forall y)[x \triangleleft^{+} y \rightarrow \neg P(y)]]\,].$$

In converting this to a first-order schema we strengthen it somewhat.[5]

Fin-D $(\forall x)[\,\phi(x) \rightarrow (\exists y)[x \triangleleft^{*} y \wedge \phi(y) \wedge (\forall z)[y \triangleleft^{+} z \rightarrow \neg\phi(z)]]\,].$

This asserts that whenever some first-order definable set includes some node, then the subset of that set that is dominated by that node will include some maximal node.

Let $\mathcal{A}_{\mathbf{Fin\text{-}D}}$ be the union of $\mathcal{A}$ and **Fin-D**. Our claim is that $\mathcal{A}_{\mathbf{Fin\text{-}D}}$ implies exactly the first-order theory of trees in which every node has finite depth. To establish it, we need to show that the first-order consequences of $\mathcal{A}_{\mathbf{Fin\text{-}D}}$ coincide with the first-order consequences of $\mathcal{A}$ plus the second-order axiom on which **Fin-D** is based.

2.3.3. *Bounded and Finite Branching*

We turn now to the issue of restricting our models to those with finite branching. One extremely simple way of doing this is to fix a finite bound on the branching factor of the trees. For binary branching, for instance, we can add the axiom:

BB2 $(\forall x)[\,(\exists y)[x \triangleleft y] \rightarrow (\exists y_1, y_2)(\forall z)[x \triangleleft z \rightarrow (z \approx y_1 \vee z \approx y_2)]\,].$

It is easy to modify this to yield axioms **BB*n*** which fix the bound at any given $n \in \mathbb{N}$. For many linguistic theories this suffices. In fact, it is a principle of some theories that such a bound exists (Kayne, 1981). For other theories, "flat" accounts

[4] To see this, assume that we are given a non-empty set S. If the root is in S, then it is, by definition, minimum. Otherwise the root is in the complement of S and is not dominated by any node in S. The set of all nodes that are not dominated by any node in S, then, is non-empty and must, by hypothesis, include a maximal node. Since the p.o is discrete, there will be a least node dominated by that maximal node. That node, by the way it is chosen, must be dominated by a member of S but is not properly dominated by any member of S. It follows that it is in S, and further, is minimal in S. Note that this argument, like the argument for the equivalence of induction and well-foundedness, is valid even if we restrict ourselves to first-order definable sets, since the property of being dominated by a node in a first-order definable set is first-order definable and the class of first-order definable sets is closed under complement.

[5] This axiom schema is adapted from Blackburn and Meyer-Viol (1994). The corresponding modification of the second-order axiom does not strengthen it. If every subset includes a maximal node then every subset of the set of nodes dominated by a given point will include a maximal node as well. The reason we employ the modified form is that it may strengthen the first-order schema. That is, the fact that every first-order definable set includes a maximal node does not suffice to guarantee that the subset dominated by any node in that set includes a maximal point, rather it only guarantees that every subset dominated by a first-order definable node in that set will include a maximal point.

of coordination, for instance, or, more generally, theories expressed in formalisms in which rewriting rules may employ regular expressions (Gazdar *et al.*, 1985), we must allow arbitrary finite branching. Here we can use a schema analogous to the one we used for finite branches, albeit simplified slightly by the fact that sets of siblings are linearly (rather than partially) ordered by left-of.

Fin-B

$$(\forall x)[\,(\exists y)[x \triangleleft y \wedge \phi(y)] \rightarrow (\exists y)[x \triangleleft y \wedge \phi(y) \wedge (\forall z)[(x \triangleleft z \wedge y \prec z) \rightarrow \neg\phi(z)]]\,].$$

This states that every definable subset of the set of children of a node has a maximum wrt linear precedence.

Let $\mathcal{A}_{\mathbf{BBn}}$ be $\mathcal{A}_{\mathbf{Fin\text{-}D}}$ augmented with $\mathbf{BB}n$ and $\mathcal{A}_{\mathbf{Fin}}$ be the union of $\mathcal{A}_{\mathbf{Fin\text{-}D}}$ with **Fin-B**. Our claims are that these axiomatize the first-order theories of finite trees with no more than binary branching and finite trees with arbitrary branching, respectively. It is these claims that we prove in the second half of this paper.

2.3.4. *A Note on the Axiomatizations*

Our basic set of axioms $\mathcal{A}$ captures the properties of trees that are usually enumerated in the linguistic literature. As we have shown, these properties, by themselves, are not sufficient to prove all properties of finite trees. In practice, of course, arguments about the structure of trees are not limited to deductions from these properties. Rather, they typically employ induction, either on the depth of nodes or possibly on the number of children preceding a node. In the case of finite trees, these might be augmented with inferences from the fact that every branch and every set of children are bounded by a maximum node. We have shown that the second-order axiom corresponding to **Fin-D** implies that domination is a well-founded partial-ordering of the nodes in the tree, and it is a well-known result that this is the case iff induction is valid. It is not hard to show, as well, that induction plus the existence of a maximum for every branch implies **Fin-D**. Similar arguments can be carried out for **Fin-B**. Consequently, rather than pointing to a gap in the foundations of these arguments about the structure of trees, our results actually confirm that the techniques generally employed in these arguments are capable, at least in principle, of deriving every first-order property of finite trees.

3. COMMON ASPECTS OF THE PROOFS

To establish that the consequences of $\mathcal{A}_{\mathbf{BBn}}$ and $\mathcal{A}_{\mathbf{Fin}}$ coincide with the first-order theory of finite trees with bounded branching and the first-order theory of all finite trees, respectively, we must show that every first-order sentence satisfied by any model of these axioms is satisfied by some intended model. One way of doing this would be to show that every model of the axioms is elementarily equivalent to an intended model, that is, for every model of the axioms there is some intended model that satisfies all and only the sentences satisfied by that model. This, however, is not the case. Every infinite model of the axioms, for example, satisfies all sentences of the form: "There are at least n distinct nodes in the tree", but every finite tree satisfies at most finitely many of them.

How, then, are we to establish our claim? All we are required to show is that every sentence satisfied by a non-standard model is satisfied by some finite tree, not that all such sentences are satisfied by the same finite tree. Note that for our example sentences (asserting the existence of n distinct nodes) it is trivially the case that each sentence is satisfied by a finite tree, although no finite tree satisfies all of them. Suppose, then, that we are given an arbitrary sentence that is satisfied by a given non-standard model. As every sentence is finite, the depth of the nesting of the quantifiers in that sentence is finite. That depth is referred to as the *quantifier rank* of the sentence.[6] The idea is to show, for any non-standard model and all n, that there is some intended model that satisfies every sentence of quantifier rank less than or equal to n that is satisfied by the given non-standard model. We say that such an intended model is n-equivalent to the non-standard model. The nature of our proofs is to exhibit a construction that, given a non-standard model and an arbitrary n, produces an intended model that is n-equivalent.

3.1. Ehrenfeucht-Fraïssé Games

A standard method (which we will use extensively) of establishing the n-equivalence of two structures uses Ehrenfeucht's game-theoretic interpretation of Fraïssé's algebraic characterization of equivalence. We sketch this here. (For a more complete introduction see Ebbinghaus *et al.*, 1984.)

Suppose ϕ is a formula of L. We define the *quantifier rank* of ϕ, $\mathrm{qr}(\phi)$, in the standard way.

DEFINITION 5 (Quantifier rank).

$$\begin{aligned}
\mathrm{qr}(\phi) &= 0 && \text{if } \phi \text{ is atomic}\\
\mathrm{qr}(\neg\phi) &= \mathrm{qr}(\phi) &&\\
\mathrm{qr}(\phi \wedge \psi) &= \max(\mathrm{qr}(\phi), \mathrm{qr}(\psi)) && \text{similarly for other connectives}\\
\mathrm{qr}(\forall x \phi) &= \mathrm{qr}(\phi) + 1 &&\\
\mathrm{qr}(\exists x \phi) &= \mathrm{qr}(\phi) + 1 &&
\end{aligned}$$

DEFINITION 6 (Restricted Languages). Let L^n denote the set of formulae in L that have quantifier rank n. Let L_k denote the set of formulae in L with k free variables. Let L^n_k denote the intersection of these sets.

Clearly L^n_k contains trivial variants of every formula in L^m_j for all $m \leq n$ and $j \leq k$.

[6] We provide a rigorous definition of this notion in the next section.

DEFINITION 7 (Logical equivalence). Two L-structures, A and B, are *elementarily equivalent* if $A \models \phi \Leftrightarrow B \models \phi$ for all sentences $\phi \in L$.

Two L-structures, A and B, are *n-equivalent* if $A \models \phi \Leftrightarrow B \models \phi$ for all sentences $\phi \in L^n$.

Ehrenfeucht's characterization of n-equivalence is based on a *pebble game* in which there are two competitors, a *duplicator* (Dup) who is seeking to demonstrate the similarity of the structures and a *spoiler* (Spo) who is seeking to show their dissimilarity. The game is played with a finite set of numbered pairs of pebbles. Spo plays first, placing a pebble on any point in the universe of either structure. Dup then replies by placing the pebble with the same number on some point in the universe of the other. Dup wins the n-pebble game if, after n rounds, the map taking pebbled points in the first structure to the points marked with the same number pebble in the other is a *partial isomorphism*. That is, if we let h be the map defined by the pebbles (taking some subset of $|A|$ into a subset of $|B|$), then h is one-to-one and preserves the constants and relations of A and B in the sense that, for all constants c and relations R interpreted by A and B,[7] letting c^A and R^A denote A's interpretation of c and R, respectively, and for all $a, b \in \delta(h)$ (the domain of h):

- $c^A = a \Leftrightarrow c^B = h(a)$, and
- $\langle a, b\rangle \in R^A \Leftrightarrow \langle h(a), h(b)\rangle \in R^B$.

We say that Dup has a *winning strategy* for the n-pebble game on A, B, if there is a fixed strategy that Dup can follow that wins against any sequence of moves by Spo. Ehrenfeucht's Theorem relates n-equivalence to the existence of a winning strategy for the n-pebble game.

THEOREM 8 (Ehrenfeucht). *If A and B are both L structures for some language L, then $A \equiv_n B$ iff Dup has a winning strategy for the n-pebble game on A, B.*

Typically, one establishes the n-equivalence of two structures by presenting a winning strategy for Dup for the n-pebble game on those structures.[8] In our proofs we will generally be establishing that various operations on structures preserve n-equivalence. In these cases we assume the existence of a winning strategy for the n-pebble game on the original structures, and show how it can be modified to yield a winning strategy for the n-pebble game on the structures resulting from application of the operation.

3.2. Types in Restricted Languages

The key observation underlying our constructions is that there are only finitely many properties of (tuples of) points that can be expressed by formulae with

[7] If our language included function symbols these would be required to be preserved as well.

[8] Or, even more typically, establishes their elementary equivalence by presenting such a strategy for arbitrary n.

bounded quantifier rank in a finite relational language. Thus, while models may well include infinitely many distinct nodes, formulae with bounded quantifier rank in our language can distinguish only finitely many classes of these. We can formalize these ideas using the standard model-theoretic notion of *types*. For a k-tuple of points in a model A, the k-type of that tuple (in A) is the set of properties that it exhibits, that is, the set of formulae that the tuple makes true in A.

DEFINITION 9 (Types). Suppose $\langle a_1, \ldots, a_k \rangle$ is a k-tuple of nodes in a model A.

The *k-type* of $\langle a_1, \ldots, a_k \rangle$ in A is the set of all formulae in k free variables that are satisfied by $\langle a_1, \ldots, a_k \rangle$ in A:

$$\mathrm{tp}_A(a_1, \ldots, a_k) \stackrel{\mathrm{def}}{=} \{\phi(x_1, \ldots, x_k) \mid A \models \phi[a_1, \ldots, a_k]\}.$$

The set of *k-types* realized in A is the set of k-types of tuples in A:

$$S_k(A) \stackrel{\mathrm{def}}{=} \left\{\mathrm{tp}_A(a_1, \ldots, a_k) \mid \langle a_1, \ldots, a_k \rangle \in |A|^k\right\}.$$

We extend this notion slightly to types restricted to formulae of bounded quantifier rank.

DEFINITION 10 (Types in L^n). Suppose $\langle a_1, \ldots, a_k \rangle$ is a k-tuple of nodes in a model A.

The *n, k-type* of $\langle a_1, \ldots, a_k \rangle$ in A is the set of sentences of quantifier rank n satisfied by $\langle a_1, \ldots, a_k \rangle$ in A:

$$\mathrm{tp}^n_A(a_1, \ldots, a_k) \stackrel{\mathrm{def}}{=} \mathrm{tp}_A(a_1, \ldots, a_k) \cap L^n.$$

The set of *n, k-types* realized in A is the set of n, k-types of the k-tuples in A:

$$S^n_k(A) \stackrel{\mathrm{def}}{=} \left\{\mathrm{tp}^n_A(a_1, \ldots, a_k) \mid \langle a_1, \ldots, a_k \rangle \in |A|^k\right\}.$$

Remark 11. $\mathrm{tp}^n_A(a_1, \ldots, a_k)$ is complete in the sense that, for all formulae in L^n_k, either that formula or its negation is in $\mathrm{tp}^n_A(a_1, \ldots, a_k)$.

If A is an L-structure with $a_1, \ldots, a_k \in A^k$, the type of $\langle a_1, \ldots, a_k \rangle$ in A can be considered to be the set of all properties definable in L that hold of this k-tuple of elements in A. The types of two k-tuples are equal, then, iff the tuples are indistinguishable by (satisfaction of) formulae in L. $S_k(A)$ is the set of types of k-tuples in A which are distinguishable by properties definable in L. When we consider properties definable in L^n (i.e., with quantifiers nested only n deep), we have the n, k-type of $\langle a_1, \ldots, a_k \rangle$ in A and $S^n_k(A)$, the set of n, k-types realized in

A. Note that for the empty tuple, ε, $\mathrm{tp}_A(\varepsilon)$ is just the set of sentences satisfied by A, that is, the theory of A.

In the following we observe that some key properties follow when we restrict the language to a finite number of relation and constant symbols, and no function symbols; a restriction satisfied by the language of our axiomatizations. For languages of this kind, the number of n, k-types realized in any L-structure (that is, the number of k-tuples of elements in a structure distinguishable by L^n) is finite and each n, k-type is characterized by a formula in L^n.

The key result is given by the following lemma, which is well known.

LEMMA 12. *For all $n, k \in \mathbb{N}$, there are but finitely many logically distinct formulae of quantifier rank n in k free variables in any finite relational language L (augmented, possibly, with finitely many constants).*

Proof. (By induction on n.) Formulae of L^0_k are just Boolean combinations of literals of L in k free variables. Since, modulo renaming of the variables, there are finitely many terms in L_k—just the variables and the finitely many constants—and since L contains only finitely many relational symbols, there are finitely many of such literals (l, say). Every Boolean combination of these has a logical equivalent that is in CNF. Since the number of literals is bounded, the number of logically distinct disjunctions of these literals is bounded (by 2^l) and the number of logically distinct conjunctions of those disjunctions is bounded (by 2^{2^l}). This establishes the lemma for $n = 0$.

For the induction step, note that formulae of L^{i+1}_k are Boolean combinations of formulae of the form $(\exists x)[\psi(x)]$ or $(\forall x)[\psi(x)]$ where $\psi(x)$ are formulae in L^i_{k+1}. If we treat formulae of this form as literals, the argument for the base case applies again here. Thus, every formula in L^{i+1}_k is logically equivalent to some conjunction of boundedly many disjunctions of boundedly many formulae in L^i_{k+1}, and the fact that there are but finitely many logically distinct formulae in L^i_{k+1} implies that there are but finitely many logically distinct formulae in L^{i+1}_k.

This lemma establishes that there are only finitely many properties of tuples of k individuals that can be expressed in L if quantifiers can be nested only n deep. That is, for every such language and $n, k \in \mathbb{N}$ there is a finite set of formulae $\Phi^{n,k}_L$ such that, for all $\psi \in L^n_k$ there exists some $\phi \in \Phi^{n,k}_L$ such that, for all L-structures A and all tuples $\langle a_1, \ldots, a_k\rangle \in |A|^k$:

$$A \models \psi[a_1, \ldots, a_k] \Leftrightarrow A \models \phi[a_1, \ldots, a_k].$$

For an L-structure, A, and $\langle a_1, \ldots, a_k\rangle \in |A|^k$, let

$$\Phi^{n,k}_{A,\langle a_1,\ldots,a_k\rangle} = \{\phi(x_1, \ldots, x_k) \mid \phi(x_1, \ldots, x_k) \in \Phi^{n,k}_L \text{ and } A \models \phi[a_1, \ldots, a_k]\}$$

Thus the set $\Phi^{n,k}_{A,\langle a_1,\ldots,a_k\rangle}$ logically implies the entire type $\mathrm{tp}^n_A(a_1, \ldots, a_k)$. As this is a subset of $\Phi^{n,k}_L$, it is finite and the conjunction of formulae in it implies the entire type. Furthermore, that conjunction is, itself, in $\mathrm{tp}^n_A(a_1,\dots,a_k)$. Thus there is a single formula in the n,k-type that is logically equivalent to the entire type.

COROLLARY 13. *For L in the class of languages we have assumed, all $n,k \in \mathbb{N}$, and every n,k-type realized in an L-structure A there is some formula*

$$\chi^n_{A,\langle a_1,\dots,a_k\rangle}(x_1,\dots,x_k) \in \mathrm{tp}^n_A(a_1,\dots,a_k)$$

such that, for all models B and $\langle b_1,\dots,b_k\rangle \in |B|^k$

$$\begin{aligned} &B \models \chi^n_{A,\langle a_1,\dots,a_k\rangle}[b_1,\dots,b_k] \Leftrightarrow \\ &\qquad B \models \psi[b_1,\dots,b_k] \text{ for all } \psi(x_1,\dots,x_k) \in \mathrm{tp}^n_A(a_1,\dots,a_k) \end{aligned}$$

The formula $\chi^n_{A,\langle a_1,\dots,a_k\rangle}(x_1,\dots,x_k)$ is just $\bigwedge \Phi^{n,k}_{A,\langle a_1,\dots,a_k\rangle}$.

It follows from the fact that the $\mathrm{tp}^n_A(a_1,\dots,a_k)$ are complete that this formula characterizes the tuples of n,k-type $\mathrm{tp}^n_A(a_1,\dots,a_k)$.

COROLLARY 14. *For L in the class of languages we have assumed and for all $n,k \in \mathbb{N}$, L-structures A, B, and tuples $\langle a_1,\dots,a_k\rangle \in |A|^k$, $\langle b_1,\dots,b_k\rangle \in |B|^k$*

$$B \models \chi^n_{A,\langle a_1,\dots,a_k\rangle}[b_1,\dots,b_k] \Leftrightarrow \mathrm{tp}^n_B(b_1,\dots,b_k) = \mathrm{tp}^n_A(a_1,\dots,a_k).$$

Since there are but finitely many logically distinct formulae that can characterize an n,k-type, there are only finitely many n,k-types that can be realized in any L-structure.

COROLLARY 15. *For L in the class of languages we have assumed and for all $n,k \in \mathbb{N}$, the set*

$$\bigcup_{A \text{ an } L\text{-structure}} S^n_k(A)$$

is finite.

Another way of focusing on the properties of (a tuple of) nodes in a model by naming them with constants.

DEFINITION 16 (Augmented models). Suppose A is an L-structure and $a \in |A|$. Let $L(c)$ denote L augmented with a *new* constant c. Then A *adjoin* a—denoted (A,a)—is an $L(c)$-structure that extends A by interpreting c as a.

The following lemma and its corollary show that we can work interchangeably with $\mathrm{tp}^n_A(\bar{a})$ and $(A,\bar{a})$. It is often easier to visualize theorems stated in terms of the augmented structures, but we generally will choose the form to suit our convenience.

LEMMA 17. $(A, a) \equiv_n (B, b) \Leftrightarrow \mathrm{tp}^n_A(a) = \mathrm{tp}^n_B(b)$.

Proof. Recall $L(\hat{a})^n$ is L augmented with a new constant ($\hat{a}$ here) restricted to formulae of quantifier rank n. By definition

$$(A, a) \equiv_n (B, b) \overset{\text{def}}{\Longleftrightarrow} \{\phi \in L(\hat{a})^n \mid (A, a) \models \phi\} = \{\phi \in L(\hat{a})^n \mid (B, b) \models \phi\}.$$

To show that the n-equivalence of (A, a) and (B, b) implies that the $n, 1$-type of a in A is the same as the $n, 1$-type of b in B, suppose $\phi(x) \in \mathrm{tp}^n_A(a)$. Let $\phi(x \mapsto \hat{a})$ be $\phi(x)$ with $\hat{a}$ uniformly substituted for x.

$$\begin{aligned} \phi(x) \in \mathrm{tp}^n_A(a) &\Leftrightarrow A \models \phi[a] \\ &\Leftrightarrow (A, a) \models \phi(x \mapsto \hat{a}) \\ &\Leftrightarrow (B, b) \models \phi(x \mapsto \hat{a}) \\ &\Leftrightarrow B \models \phi[b] \\ &\Leftrightarrow \phi(x) \in \mathrm{tp}^n_B(b). \end{aligned}$$

For the other direction, suppose $\phi \in L(\hat{a})^n$.

$$\begin{aligned} (A, a) \models \phi &\Leftrightarrow A \models \phi(\hat{a} \mapsto x)[a] \\ &\Leftrightarrow \phi(\hat{a} \mapsto x) \in \mathrm{tp}^n_A(a) \\ &\Leftrightarrow \phi(\hat{a} \mapsto x) \in \mathrm{tp}^n_B(b) \\ &\Leftrightarrow B \models \phi(\hat{a} \mapsto x)[b] \\ &\Leftrightarrow (B, b) \models \phi. \end{aligned}$$

The above lemma can be generalized to the case when L is augmented with any finite number of constants.

COROLLARY 18. *For all $k \in \mathbb{N}$, $\bar{a} \in |A|^k$, and $\bar{b} \in |B|^k$*

$$(A, \bar{a}) \equiv_n (B, \bar{b}) \Leftrightarrow \mathrm{tp}^n_A(\bar{a}) = \mathrm{tp}^n_B(\bar{b}).$$

This follows by induction on k, since we can take A and B in the lemma to be models with adjoined points.

By combining Corollaries 18 and 14, we have the following.

COROLLARY 19. *For L in the class of languages we have assumed and for all $n, k \in \mathbb{N}$, L-structures A, B, and tuples $\langle a_1, \ldots, a_k \rangle \in |A|^k$, $\langle b_1, \ldots, b_k \rangle \in |B|^k$*

$$B \models \chi^n_{A,\langle a_1,\ldots,a_k\rangle}[b_1, \ldots, b_k] \Leftrightarrow (B, b_1, \ldots, b_k) \equiv_n (A, a_1, \ldots, a_k).$$

A case of particular interest to us in our constructions is the case of trees (or, more generally, models of our axioms) in which the root has been distinguished by a constant.

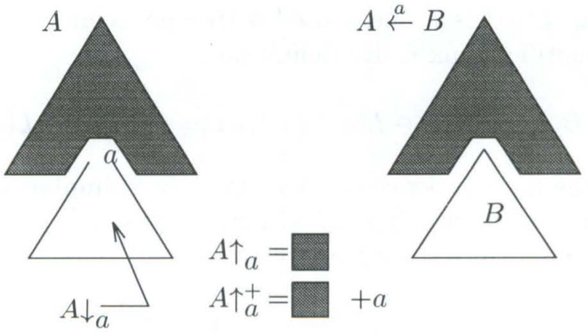

Fig. 2. Subtrees and substitution.

COROLLARY 20. *For L in the class of languages we have assumed and all $n \in \mathbb{N}$, and L-structures A, B*

$$B \models \chi^n_{A,\langle \mathrm{r}(A)\rangle}[\mathrm{r}(B)] \Leftrightarrow (B, \mathrm{r}(B)) \equiv_n (A, \mathrm{r}(A)).$$

3.3. Some Operations on Models of $\mathcal{A}$

As we noted earlier, to show that our axioms imply all properties of finite trees, we will show that each sentence consistent with the axioms is satisfied by some intended model. The nature of our proofs is to take an arbitrary model of the axioms that satisfies a given sentence, and to construct from that model an intended model that satisfies the same sentence. We do this by deleting all but finitely much of the original model while preserving satisfaction of the given sentence and of the axioms. In this section we introduce the basic operations that we employ in these constructions. These isolate or delete certain sub-models, models built on subsets of the universe of original model.

DEFINITION 21 (Restrictions of models). Suppose $A = \langle |A|, \mathcal{I}, \mathcal{D}, \mathcal{P}\rangle$ is a structure and $X \subseteq |A|$. Then the *restriction of A to X* is:

$$A|_X \overset{\mathrm{def}}{=} \left\langle X, \mathcal{I} \cap X^2, \mathcal{D} \cap X^2, \mathcal{P} \cap X^2 \right\rangle.$$

DEFINITION 22 (Subtrees). Given a structure $A = \langle |A|, \mathcal{I}, \mathcal{D}, \mathcal{P}\rangle$ and a node $a \in |A|$, let $|A{\downarrow}_a| \overset{\mathrm{def}}{=} \{b \mid \langle a, b\rangle \in \mathcal{D}\}$. Then the *subtree of A at a* is:

$$A{\downarrow}_a \overset{\mathrm{def}}{=} A|_{|A{\downarrow}_a|},$$

Suppose $A = \langle |A|, \mathcal{I}, \mathcal{D}, \mathcal{P} \rangle$, and $a \in |A|$. Let $|A{\uparrow}_a| \overset{\text{def}}{=} |A| \setminus |A{\downarrow}_a|$. The *subtree of A above a* is:

$$A{\uparrow}_a \overset{\text{def}}{=} A|_{|A{\uparrow}_a|},$$

and the *subtree of A not below a* is:

$$A{\uparrow}_a^+ \overset{\text{def}}{=} A|_{|A{\uparrow}_a| \cup \{a\}}.$$

Note $a \notin |A{\uparrow}_a|$ but $a \in |A{\uparrow}_a^+|$.

We can characterize the subtrees of a model in much the same way as we characterize the $n, 1$-types of individual nodes.

LEMMA 23. *Suppose A is an L-structure and $a \in |A|$. Then there is an L_k^n-formula $\tau_{A,a}^n(x)$ such that*

$$A \models \tau_{A,a}^n[b] \Leftrightarrow (A{\downarrow}_b, b) \equiv_n (A{\downarrow}_a, a).$$

Proof. By Corollary 20 there is a formula $\chi_{A{\downarrow}_a,\langle a \rangle}^n(x)$ such that

$$A{\downarrow}_b \models \chi_{A{\downarrow}_a,\langle a \rangle}^n[b] \Leftrightarrow (A{\downarrow}_b, b) \equiv_n (A{\downarrow}_a, a).$$

Let $\tau_{A,a}^n(x)$ be $\chi_{A{\downarrow}_a,\langle a \rangle}^n(x)$ relativized to x by replacing every instance of $(\forall y)[\phi(y)]$ with $(\forall y)[x \lhd^* y \rightarrow \phi(y)]$ and every instance of $(\exists y)[\phi(y)]$ with $(\exists y)[x \lhd^* y \wedge \phi(y)]$. All quantification in $\tau_{A,a}^n(x)$ is restricted to nodes dominated by x. It is easy to see, then, that

$$A \models \tau_{A,a}^n[b] \Leftrightarrow A{\downarrow}_b \models \chi_{A{\downarrow}_a,\langle a \rangle}^n[b],$$

and, equivalently, $(A{\downarrow}_b, b) \equiv_n (A{\downarrow}_a, a)$.

DEFINITION 24 (Substitutions). Given the two structures $A = \left\langle |A|, \mathcal{I}^A, \mathcal{D}^A, \mathcal{P}^A \right\rangle$ and $B = \left\langle |B|, \mathcal{I}^B, \mathcal{D}^B, \mathcal{P}^B \right\rangle$ and a node $a \in |A|$, the *substitution of B at a in A* is:

$$A \overset{a}{\leftarrow} B \overset{\text{def}}{=} \langle \mathcal{U}', \mathcal{I}', \mathcal{D}', \mathcal{P}' \rangle$$

where (using $\uplus$ to denote disjoint union):

$$\begin{aligned}
\mathcal{U}' &\overset{\text{def}}{=} (|A| \setminus |A{\downarrow}_a|) \uplus |B| \\
\mathcal{I}' &\overset{\text{def}}{=} (\mathcal{I}^A \uplus \mathcal{I}^B \cup \left\{\langle a', \mathrm{r}(B) \rangle \mid \langle a', a \rangle \in \mathcal{I}^A \right\}) \cap (\mathcal{U}')^2 \\
\mathcal{D}' &\overset{\text{def}}{=} (\mathcal{D}^A \uplus \mathcal{D}^B \cup \left\{\langle a', b \rangle \mid \langle a', a \rangle \in \mathcal{D}^A, b \in |B| \right\}) \cap (\mathcal{U}')^2 \\
\mathcal{P}' &\overset{\text{def}}{=} (\mathcal{U}')^2 \cap (\mathcal{P}^A \uplus \mathcal{P}^B \cup \\
&\qquad \left\{\langle c, d \rangle \mid d \in |B| \text{ and } \langle c, a \rangle \in \mathcal{P}^A \text{ or } c \in |B| \text{ and } \langle a, d \rangle \in \mathcal{P}^A \right\}).
\end{aligned}$$

Note that we take disjoint unions when forming the new structure. This is necessary to ensure that the operation preserves satisfaction of our axioms. Note also, that in this definition a is not in the result of substituting B at a, rather it has been replaced with the root of B ($\mathrm{r}(B)$). These operations are depicted diagrammatically in Figure 2.

Under appropriate conditions, substitution can be generalized to the case of multiple simultaneous substitutions. If $\bar{a} = \langle a_i \mid i < l \rangle$ is a sequence of points in $|A|$ that are pairwise incomparable wrt domination, and $\mathbf{B} = \langle B_i \mid i < l \rangle$ is a sequence of models, then the simultaneous substitution of $\mathbf{B}$ at $\bar{a}$ in A is:

$$A \xleftarrow{\bar{a}} \mathbf{B} \overset{\text{def}}{=} A \overset{a_0}{\leftarrow} B_0 \overset{a_1}{\leftarrow} B_1 \cdots \overset{a_i}{\leftarrow} B_i \cdots, \text{ for all } i < l.$$

It is a lemma that the fact that $\bar{a}$ is pairwise incomparable wrt domination ensures that the order of the substitutions is irrelevant.

We can extend the notions of subtrees and substitutions to augmented models as well. In particular $(A, \bar{a})\!\uparrow_a = (A, \bar{a}')\!\uparrow_a$, where $\bar{a}'$ is just the subsequence of $\bar{a}$ that contains all and only those points in $\bar{a}$ that are not dominated by a. Similarly for $(A, \bar{a})\!\downarrow_a$ and $(A, \bar{a})\!\uparrow^+_a$ (using the appropriate subsequence $\bar{a}'$). The substitution $(A, \bar{a}) \overset{a}{\leftarrow} (B, \bar{b})$ is taken to be $(A \overset{a}{\leftarrow} B, \bar{a}', \bar{b})$.

LEMMA 25. *Satisfaction of the axioms is preserved under substitutions and restriction to subtrees.*

That is, the result of applying these operations to models of our axioms will also be models of those axioms.

With the next lemma we establish that n-equivalence is a congruence wrt substitution in the sense that if two models with distinguished nodes are n-equivalent, then the substitution of two n-equivalent models (with distinguished roots) at those nodes will also be n-equivalent.

LEMMA 26 (Congruence). *If* $(A, a) \equiv_n (B, b)$ *and* $(C, \mathrm{r}(C)) \equiv_n (D, \mathrm{r}(D))$, *then* $(A \overset{a}{\leftarrow} C, \mathrm{r}(C)) \equiv_n (B \overset{d}{\leftarrow} D, \mathrm{r}(D))$.

Proof. We claim that the combination of Dup's strategy for the n-pebble game on $(A, a), (B, b)$ with Dup's strategy for $(C, \mathrm{r}(C)), (D, \mathrm{r}(D))$ serves as a winning strategy for $(A \overset{a}{\leftarrow} C, \mathrm{r}(C))$, $(B \overset{d}{\leftarrow} D, \mathrm{r}(D))$. (Note that the strategy covers $|A\!\downarrow_a|$ and $|B\!\downarrow_b|$ as well, but these never come into play, since none of these points are in the universes of $A \overset{a}{\leftarrow} C$ or $B \overset{b}{\leftarrow} D$.) To establish this, we need to show that the union of partial isomorphisms constructed by these strategies is a partial isomorphism from $A \overset{a}{\leftarrow} C$ to $B \overset{b}{\leftarrow} D$. Since the domains and ranges of these partial isomorphisms are disjoint, their union is a well-defined map $\left|A \overset{a}{\leftarrow} C\right| \to \left|B \overset{b}{\leftarrow} D\right|$. Further, they certainly preserve relations between points occurring only in $A\!\uparrow_a$, only in $B\!\uparrow_b$, only in C, or only in D. We need only to show that they preserve relations between pairs of points drawn from separate regions of the structures.

Let h be the union of a pair of partial maps as above. Suppose $a', c' \in \delta(h)$, and that $a' \in \left|A \overset{a}{\leftarrow} C\!\uparrow_{\mathrm{r}(C)}\right|$ and $c' \in \left|A \overset{a}{\leftarrow} C\!\downarrow_{\mathrm{r}(C)}\right|$. Then $\langle \mathrm{r}(C), a' \rangle \notin \mathcal{D}^{A \overset{a}{\leftarrow} C}$

and $\langle \mathrm{r}(C), c' \rangle \in \mathcal{D}^{A \overset{a}{\leftarrow} C}$. Further, since h necessarily maps the regions above and below $\mathrm{r}(C)$ in $A \overset{a}{\leftarrow} C$ to the corresponding regions of $B \overset{b}{\leftarrow} D$, we have that $\langle \mathrm{r}(D), h(a') \rangle \notin \mathcal{D}^{B \overset{b}{\leftarrow} D}$ and $\langle \mathrm{r}(D), h(c') \rangle \in \mathcal{D}^{B \overset{b}{\leftarrow} D}$.

Then:

$$\begin{aligned}
\langle a', c' \rangle \in \mathcal{I}^{A \overset{a}{\leftarrow} C} &\Leftrightarrow \langle a', a \rangle \in \mathcal{I}^{A} \text{ and } c' = \mathrm{r}(C) \\
&\Leftrightarrow \langle h(a'), b \rangle \in \mathcal{I}^{B} \text{ and } h(c') = \mathrm{r}(D) \\
&\Leftrightarrow \langle h(a'), h(c') \rangle \in \mathcal{I}^{B \overset{b}{\leftarrow} D}. \\
\langle a', c' \rangle \in \mathcal{D}^{A \overset{a}{\leftarrow} C} &\Leftrightarrow \langle a', a \rangle \in \mathcal{D}^{A} \\
&\Leftrightarrow \langle h(a'), b \rangle \in \mathcal{D}^{B} \\
&\Leftrightarrow \langle h(a'), \mathrm{r}(D) \rangle \in \mathcal{D}^{B \overset{b}{\leftarrow} D} \\
&\Leftrightarrow \langle h(a'), h(c') \rangle \in \mathcal{D}^{B \overset{b}{\leftarrow} D}.
\end{aligned}$$

The cases of $\langle a', c' \rangle \in \mathcal{P}^{A \overset{a}{\leftarrow} C}$ and $\langle c', a' \rangle \in \mathcal{P}^{A \overset{a}{\leftarrow} C}$ are similar to $\langle a', c' \rangle \in \mathcal{D}^{A \overset{a}{\leftarrow} C}$.

4. PROOFS OF THE COMPLETENESS OF THE AXIOMS

We now turn to proving that the first-order consequences of our axioms coincide with the first-order theory of finite trees (with bounded and arbitrary branching, respectively). We will follow the pattern of our development of the axioms and focus first on the issue of non-standard models with infinite depth. To this end, we consider first, in the next section, models in which branching is bounded by a constant. We show that the set $\mathcal{A}_{\mathbf{BBn}}$ (consisting of the basic axioms of Section 2.1, the schema **Fin-D** of Section 2.3.2, and the axiom **BB*n*** of Section 2.3.3) implies every sentence that is satisfied by every finite tree in which no node has more than n children. This is done by showing that every sentence that is satisfied by any model of the axioms, in particular by any non-standard model, is also satisfied by a finite tree of the appropriate type. Having established that, we will proceed, in Section 4.2, to account for trees with arbitrary finite branching. We do this by extending the proof of the bounded branching case to show that the consequences of set $\mathcal{A}_{\mathbf{Fin}}$ (consisting of $\mathcal{A}$, the schema **Fin-D**, and the schema **Fin-B** of Section 2.3.3) are exactly the first-order theory of finite trees with arbitrary branching.

4.1. Finite Trees with Bounded-Branching—Vertical Collapsing

We must show that every sentence that is satisfied by some model of the axioms $\mathcal{A}_{\mathbf{BBn}}$ is satisfied by some finite tree with at most n-ary branching. Suppose that

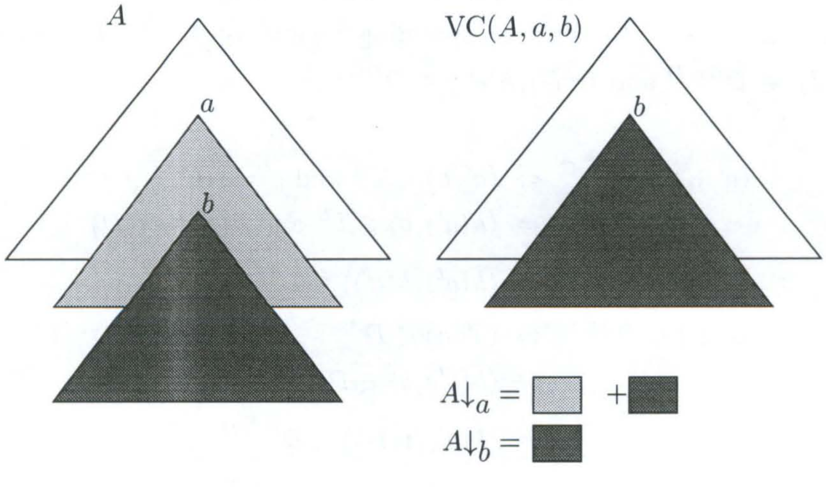

Fig. 3. Vertical collapsing.

we are given some such sentence ϕ. Let A be a model of $\mathcal{A}_{\mathbf{BBn}}$ that satisfies ϕ. Assume A is non-standard. Let n be the quantifier rank of ϕ. To show that ϕ is satisfied by an intended model, we will construct, from A, a finite tree with at most binary branching that is n-equivalent to A, and which, consequently, must satisfy ϕ. We do this by applying a sequence of substitutions which we refer to as *vertical collapsing*.

DEFINITION 27 (Vertical Collapsing). Let A be an L-structure and $a, b \in |A|$ be two nodes such that $\langle a, b\rangle \in \mathcal{D}^A$. Then the *vertical collapsing* of A at $\langle a, b\rangle$, denoted by $\mathrm{VC}(A, a, b)$, is given by $A \overset{a}{\leftarrow} A{\downarrow}_b$.

Note that vertical collapsing is defined only when the one node dominates the other. This operation is diagrammatically depicted in Figure 3. From the definition it follows that $A = \mathrm{VC}(A, a, a)$ and $|\mathrm{VC}(A, a, b)| \subseteq |A|$.

Using congruence, we can establish that if we collapse at pairs that are roots of n-equivalent subtrees in a model then the types of the subtrees of the model will be preserved.

LEMMA 28. *Suppose A is an L-structure that is a model of $\mathcal{A}$. Suppose a and b are nodes in $|A|$ such that $\langle a, b\rangle \in \mathcal{D}^A$ and $(A{\downarrow}_a, a) \equiv_n (A{\downarrow}_b, b)$. Let $A' = \mathrm{VC}(A, a, b)$. Then $A'{\downarrow}_{a'} \equiv_n A{\downarrow}_{a'}$ for all $a' \in |A'|$.*

Proof. As $|A'| \subseteq |A|$ and both are models of $\mathcal{A}$, we know, for all $a' \in |A'|$, that $a' \in |A|$ and that either $\langle b, a'\rangle \in \mathcal{D}^{A'}$, $\langle b, a'\rangle \in \mathcal{P}^{A'}$, $\langle a', b\rangle \in \mathcal{D}^{A'}$, or $\langle a', b\rangle \in \mathcal{P}^{A'}$.

Now if $\langle b, a'\rangle \in \mathcal{D}^{A'}$, $\langle b, a'\rangle \in \mathcal{P}^{A'}$, or $\langle a', b\rangle \in \mathcal{P}^{A'}$, by definition of A', we have $A'{\downarrow}_{a'} = A{\downarrow}_{a'}$ and thus the result.

The only case that remains is when $a' \neq b$ and $\langle a', b\rangle \in \mathcal{D}^{A'}$.

$$\begin{array}{lrcll}
\text{Note} & A{\downarrow}_{a'} & = & A{\downarrow}_{a'} \overset{a}{\leftarrow} A{\downarrow}_{a} & \\
 & A'{\downarrow}_{a'} & = & A{\downarrow}_{a'} \overset{a}{\leftarrow} A{\downarrow}_{b} & \\
 & (A{\downarrow}_{a}, a) & \equiv_n & (A{\downarrow}_{b}, b) & \text{by assumption} \\
\text{Trivially,} & (A{\downarrow}_{a'}, a) & \equiv_n & (A{\downarrow}_{a'}, a) & \\
\text{Hence,} & (A{\downarrow}_{a'} \overset{a}{\leftarrow} A{\downarrow}_{a}, a) & \equiv_n & (A{\downarrow}_{a'} \overset{a}{\leftarrow} A{\downarrow}_{b}, b) & \text{by congruence} \\
\text{i.e.,} & (A{\downarrow}_{a'}, a) & \equiv_n & (A'{\downarrow}_{a'}, b) & \\
\text{and thus,} & A'{\downarrow}_{a'} & \equiv_n & A{\downarrow}_{a'} &
\end{array}$$

Since this holds for the case in which a' is the root of A' we get that the result of vertically collapsing A at a pair of nodes that dominate n-equivalent subtrees is n-equivalent to A.

COROLLARY 29. *Let A be an L-structure that is a model of $\mathcal{A}$. Let $\langle a, b\rangle \in \mathcal{D}^A$ such that $(A{\downarrow}_a, a) \equiv_n (A{\downarrow}_b, b)$. Then* $\mathrm{VC}(A, a, b) \equiv_n A$.

Proof. If the root of A' is not b (i.e., if we have not collapsed at the root) then it is the root of A as well and the corollary follows from the lemma. If, on the other hand, the root of A' is b then the root of A is a and the corollary follows from the hypothesis.

The idea now is to construct a finite sequence of models starting with A in which each model is derived from its predecessor by vertical collapsing at pairs of points that dominate subtrees that are n-equivalent, and to do this in such a way that all but finitely much of the universe of the model is eventually deleted. The final tree of this sequence will be finite and, since the collapsings all satisfy the conditions of Corollary 29, it will be n-equivalent to A.

The construction proceeds in stages. Let us say that the root of a model is at depth 0, and that if a node is at depth k then its children are at depth $k+1$. At stage i the construction will focus on the nodes at depth i.

Recall from Lemma 23 that we have an L_k^n-formula $\tau^n_{A,a}(x)$ that characterizes the $n,1$-type of a in the subtree rooted at a in a model A. Let Fin-D$(\tau^n_{A,a})$ be the instance of **Fin-D**:

$$(\forall x)[\,\tau^n_{A,a}(x) \rightarrow (\exists y)[x \lhd^* y \wedge \tau^n_{A,a}(y) \wedge (\forall z)[y \lhd^+ z \rightarrow \neg\tau^n_{A,a}(z)]]\,].$$

Stage 0 of the construction

Suppose A is a model of $\mathcal{A}_{\mathbf{BBn}}$. Let $A_0 = A$ and let a_0 be the root of A. As $A_0 \models$ Fin-D$(\tau^n_{A_0,a_0})$, an instance of **Fin-D**, and $A_0 \models \tau^n_{A_0,a_0}[a_0]$, we know there is a maximal $b_0 \in |A_0|$ that is dominated by a_0 for which $A_0 \models \tau^n_{A_0,a_0}[b_0]$. In words, there is a node b_0 that is dominated by a_0 such that the type of the subtree rooted at b_0 is the same as the type of the subtree rooted at a_0 and there is no subtree of this type rooted at a node properly dominated by b_0. Formally,

$$(A_0, a_0) = (A_0{\downarrow}_{a_0}, a_0) \equiv_n (A_0{\downarrow}_{b_0}, b_0)$$

and

$$(A_0{\downarrow}_{a_0}, a_0) \not\equiv_n (A_0{\downarrow}_b, b) \text{ for all } b \in \left|A_0{\downarrow}_{b_0}\right|.$$

Let $A_1 = \mathrm{VC}(A_0, a_0, b_0)$.

Stage $i \geq 1$ of construction
We consider the nodes at depth i in A_i. As we are considering models of $\mathcal{A}_{\mathbf{BBn}}$, there are at most n^i nodes at depth i in such a model. Let these nodes be $a_{i,1}, \ldots, a_{i,m_i}$ where $0 \leq m_i \leq n^i$. As in stage 1, for each $a_{i,j}$ $(0 < j \leq m_i)$, we find a maximal $b_{i,j}$ such that $(A_i{\downarrow}_{a_{i,j}}, a_{i,j}) \equiv_n (A_i{\downarrow}_{b_{i,j}}, b_{i,j})$ by considering an appropriate instance of **Fin-D**. Let $A_{i,0} = A_i$, and, for $0 < j \leq m_i$, let $A_{i,j} = \mathrm{VC}(A_{i,j-1}, a_{i,j}, b_{i,j})$. Note that, since the $a_{i,j}$ are siblings, each of the $a_{i,k}$, for $k > j$, and every $b_{i,k}$ is in the universe of $A_{i,j}$. Lemma 28 ensures that the subtrees rooted at $a_{i,k}$ and $b_{i,k}$ in $A_{i,j}$ will still be n-equivalent. Let $A_{i+1} = A_{i,m_i}$.

Our claim is that this construction terminates after finitely many stages, that the final model is a finite tree and that it is n-equivalent to A_0.

To establish finite termination, we show that each stage of the construction reduces, by at least one, the number of distinct types of subtrees occurring below the nodes at the corresponding level. Since there can only be finitely many such distinct types in the tree to begin with, this can be repeated only finitely many times.

DEFINITION 30. Let A be a model and $a \in |A|$.

$$\text{Subtree-types}^n(A, a) \stackrel{\text{def}}{=} \left\{\mathrm{tp}^n_{A{\downarrow}_b}(b) \mid \langle a, b\rangle \in \mathcal{D}^A\right\}.$$

That is, Subtree-types$^n(A, a)$ is the set of the types of the subtrees rooted at nodes dominated by a in A (more precisely, the set of $n,1$-types of the nodes dominated by a in the subtrees rooted at those nodes). By Corollary 15 this set is always finite. Furthermore, since every node dominates at least the subtree rooted at itself, it is never empty.

LEMMA 31 (Invariant). *Let* $l = \mathbf{card}(\text{Subtree-types}^n(A_0, a_0))$. *For all* A_i *and all* b *at depth* i *in* $|A_i|$*:*

1. $\mathbf{card}(\text{Subtree-types}^n(A_i, b)) \leq l - i$.
2. $A_i \equiv_n A_0$.

Proof. This can be shown by induction on i. Clearly the invariant is true for A_0. Suppose that the invariant holds for all $j < i$. For all $i > 0$, A_i is formed at stage $i-1$ by vertically collapsing at the nodes at depth $i-1$ in A_{i-1}. That is, the nodes at depth $i-1$ in A_{i-1} are the $a_{i-1,j}$ and the nodes at depth $i-1$ in A_i are the $b_{i-1,j}$. By Lemma 28, the types of the subtrees dominated by $b_{i-1,j}$ in A_i are the same as their types in A_{i-1}. By the induction hypothesis no $b_{i-1,j}$ dominates more than $l-(i-1)$ distinct types of subtree, since these are all subtrees dominated by $a_{i-1,j}$ in A_{i-1}. Each node b at depth i in A_i is the child of some $b_{i-1,j}$. By choice of the $b_{i-1,j}$, the node b does not dominate any subtree with the same type as that rooted at $b_{i-1,j}$. It follows that the set of types of the subtrees dominated by such a b is a proper subset of the set of types of the subtrees dominated by its parent. (It does not include the type of the subtree rooted at that parent.) Thus

$$\mathbf{card}(\text{Subtree-types}^n(A_i,b)) \leq \mathbf{card}(\text{Subtree-types}^n(A_i,b_{i-1,j})) - 1 \leq l - i.$$

Finally, the n-equivalence of A_i and A_{i-1} follows from Corollary 29, and the second part of the invariant then follows by transitivity of equivalence.

From this lemma it follows that any node at depth $l-1$ in A_{l-1} must be a leaf, as no node it properly dominates could dominate any subtree at all. Consequently, there can be at most l stages in the construction and the result of the final stage is a model that is n-equivalent to A_0 in which no node is at depth greater than $l-1$. The construction, then, terminates and yields the required tree.

LEMMA 32. *For each model, A, of $\mathcal{A}_{\mathbf{BBn}}$ and each n, there is a finite-depth tree with bounded branching that is n-equivalent to A.*

This establishes our desired result, that every sentence satisfied by some model of $\mathcal{A}_{\mathbf{BBn}}$ is satisfied by a finite tree with at most n-ary branching, and therefore, that the consequences of $\mathcal{A}_{\mathbf{BBn}}$ are exactly the first-order theory of finite trees with at most n-ary branching.

LEMMA 33. *For any sentence ψ in L, if ψ is consistent with $\mathcal{A}$, $\mathbf{BB}n$, and all instances of* **Fin-D**, *then ψ is satisfied in some finite tree with at most n-ary branching.*

THEOREM 34. *The first-order consequences of $\mathcal{A}_{\mathbf{Fin}}$ are exactly the first-order theory of finite trees with at most n-ary branching.*

4.2. Finite Trees with Arbitrary Branching—Horizontal Collapsing

In the previous section, we employed vertical collapsing to construct finite-depth trees that satisfy a sentence consistent with $\mathcal{A}_{\mathbf{BBn}}$. Since $\mathbf{BB}n$ provides a finite

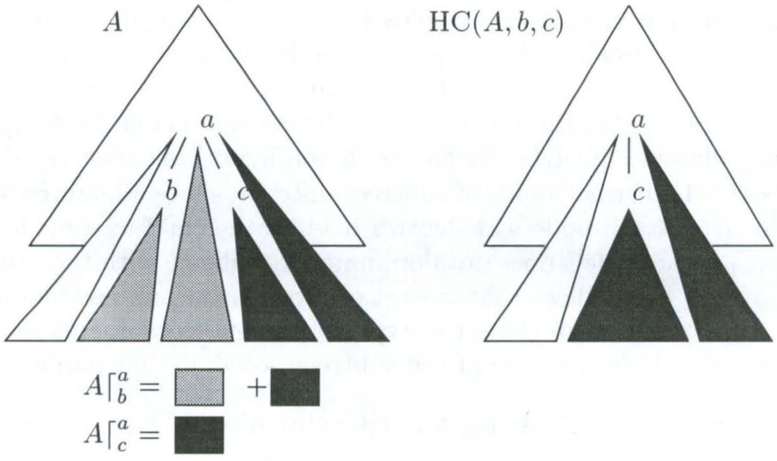

Fig. 4. Horizontal collapsing.

bound on the number of children of any node, finiteness of the depth of these trees suffices to establish finiteness of the entire tree. In this section, we replace **BB*n*** by instances of the schema **Fin-B**, and use a sequence of *horizontal collapsings* to construct models in which nodes may have any finite number of children.

We first define the horizontal collapsing operation and then show that given a model A and a node $a \in |A|$ there is a model A' obtained from A in which a has but a finite number of children. We show that A' preserves the invariants of Lemma 31, and that we, therefore, can use horizontal collapsing at each stage of the vertical collapsing construction to ensure that there are only finitely many nodes at the corresponding depth in the model.

DEFINITION 35. If A is an L-structure and $a, b \in |A|$ such that $\langle a, b\rangle \in \mathcal{I}^A$, let

$$A\lceil^a_b \stackrel{\text{def}}{=} \left\{c \mid \langle a, c\rangle \in \mathcal{D}^A \text{ and } \langle c, b\rangle \notin \mathcal{P}^A\right\}.$$

That is, when a is the parent of b, then $A\lceil^a_b$ is the set of nodes that includes a, the nodes dominated by b (which includes b as domination is a taken to be reflexive) as well as nodes dominated by the right-siblings of b. See Figure 4.

DEFINITION 36 (Horizontal Collapsing). If A is an L-structure and $b, c \in |A|$ are siblings with $\langle a, b\rangle, \langle a, c\rangle \in \mathcal{I}^A$ and $\langle b, c\rangle \in \mathcal{P}^A$, then the *horizontal collapse* of A at b and c is

$$\mathrm{HC}(A, b, c) \stackrel{\text{def}}{=} A|_{|A| \setminus (A\lceil^a_b \setminus A\lceil^a_c)}.$$

This operation is depicted pictorially in Figure 4. Horizontal collapsing is defined only at nodes that are siblings. Note that $(A\lceil^a_b \setminus A\lceil^a_c)$ is the set of descendants of a that are dominated by b or are to the right of b but left of c, and that horizontal collapsing yields a model that deletes these nodes.

In the finite-depth, bounded branching case we used vertical collapsing of a model A at a and b such that a dominated b in A and $(A{\downarrow}_a, a) \equiv_n (A{\downarrow}_b, b)$. In the current case, in addition to similar vertical collapsings, we consider the horizontal collapsing of A at b and c, where the two nodes are siblings (and whose parent is some node, say a) such that $(A, a, b) \equiv_n (A, a, c)$.[9] In constructing the required finite-tree, we will apply a sequence of collapsings that mixes horizontal and vertical collapsing. To show that horizontal collapsing does not interfere and negate the invariants of the finite-depth construction, we show the following lemma.

LEMMA 37. *Suppose A is an L-structure and $a, b, c \in |A|$ such that b and c are children of a (i.e., $\langle a, b\rangle, \langle a, c\rangle \in \mathcal{I}^A$), b is left-of c (i.e., $\langle b, c\rangle \in \mathcal{P}^A$),. and $(A, a, b) \equiv_n (A, a, c)$. Let $A' = \mathrm{HC}(A, b, c)$. Then $A'{\downarrow}_{a'} \equiv_n A{\downarrow}_{a'}$ for all $a' \in |A'|$.*

Proof. The proof is similar to the proof of the analogous lemma for vertical collapsing. The result is trivial for all nodes $a' \in |A'|$ that don't dominate a, as in such cases $A'{\downarrow}_{a'} = A{\downarrow}_{a'}$. To establish this for nodes in $|A'|$ that dominate a, we will establish it first for a itself. The result for all other nodes dominating a will then follow by the congruence lemma.

The n-equivalence of (A, a, b) and (A, a, c) is witnessed by a winning strategy for Dup for the n-pebble game on these structures. Note that every partial isomorphism constructed by this strategy will necessarily map points in $A\lceil^a_b$ to those in $A\lceil^a_c$. We form a composite strategy for the n-pebble game on $A{\downarrow}_a$, $A'{\downarrow}_a$, where $A' = \mathrm{HC}(A, b, c)$. Note that

$$|A{\downarrow}_a| \setminus A\lceil^a_b = |A'{\downarrow}_a| \setminus A\lceil^a_c.$$

For all Spo choices in this set Dup chooses the identical node. Note also that

$$(A'{\downarrow}_a)\lceil^a_c = A'\lceil^a_c = A\lceil^a_c = (A{\downarrow}_a)\lceil^a_c.$$

For all Spo choices in $A{\downarrow}_a\lceil^a_b$ or $A'{\downarrow}_a\lceil^a_c$ Dup follows the strategy on (A, a, b),(A, a, c).

Once again it is easy to show that the maps constructed by the composite strategy are functional, 1-1, and preserve relations, and are thus partial isomorphisms. Thus the composite strategy witnesses the n-equivalence of $A{\downarrow}_a$ and $A'{\downarrow}_a$.

Now for all other nodes in $|A'|$ dominating a the result follows from the fact that, by the congruence lemma, the result of substituting $A'{\downarrow}_a$ into a submodel of A for $A{\downarrow}_a$ is n-equivalent to that submodel.

As the roots of $\mathrm{HC}(A, b, c)$ and A are the same we have, as a corollary, that the model obtained after such a horizontal collapsing is n-equivalent to the original model.

[9] We consider (A, a, b) and (A, a, c) rather than (A, b) and (A, c), as it simplifies our proof.

COROLLARY 38. *Suppose A is an L-structure and $a, b, c \in |A|$ such that b and c are children of a (i.e., $\langle a, b\rangle, \langle a, c\rangle \in \mathcal{I}^A$), b is left-of c (i.e., $\langle b, c\rangle \in \mathcal{P}^A$), and $(A, a, b) \equiv_n (A, a, c)$. Then* $\mathrm{HC}(A, b, c) \equiv_n A$.

As in the vertical collapsing construction our horizontal collapsing construction involves, at each stage, a number of collapses taken in sequence. In the vertical collapsing case, the analog of Lemma 37 suffices to ensure that these operations do not interfere with each other. In this case, however, we will need a slightly stronger result, namely that, under the hypothesis of Lemma 37, horizontal collapsing at b and c does not affect the $n, 2$-types (with a) of siblings to the left of b.

LEMMA 39. *Suppose A is an L-structure and $a, b, c \in |A|$ such that b and c are children of a (i.e., $\langle a, b\rangle, \langle a, c\rangle \in \mathcal{I}^A$), b is left-of c (i.e., $\langle b, c\rangle \in \mathcal{P}^A$), and $(A, a, b) \equiv_n (A, a, c)$. Let A' be the model resulting from a horizontal collapse of A at b and c, i.e., $A' = \mathrm{HC}(A, b, c)$. Suppose, further, that $b' \in |A'|$ but $b' \notin A'\lceil^a_c$. Then $\mathrm{tp}^n_A(a, b') = \mathrm{tp}^n_{A'}(a, b')$, i.e., $(A, a, b') \equiv_n (A', a, b')$.*

Proof. To show $(A, a, b') \equiv_n (A', a, b')$, we use Ehrenfeucht games again. We claim that the strategy of Lemma 37 serves for the n-pebble game, in this case on (A, a, b'), (A', a, b'), and again this is nearly an immediate consequence of the fact that the strategy builds identity maps on nodes not in $A\lceil^a_b$ (including b') and that the relationship, in A, of b' with any node in $A\lceil^a_b$ is the same as the relationship, in A, of b' with b. This, in turn, is the same as the relationship, in $\mathrm{HC}(A, b, c)$, of b' with c; which is the same as the relationship, in $\mathrm{HC}(A, b, c)$, of b' with all nodes in $\mathrm{HC}(A, b, c)\lceil^a_c$.

Note in particular that if b' is a left-sibling of c in $A' = \mathrm{HC}(A, b, c)$ (that is, if $\langle b', c\rangle \in \mathcal{P}^{A'}$ and $\langle a, b'\rangle \in \mathcal{I}^{A'}$) and hence a left-sibling of b in A, then $b' \notin A'\lceil^a_c$. Hence, by the above lemma, we have $\mathrm{tp}^n_A(a, b') = \mathrm{tp}^n_{A'}(a, b')$.

We can now show how to construct, for any n and any model of $\mathcal{A}_{\mathbf{Fin}}$, an n-equivalent model that is isomorphic to the natural interpretation of a finite-depth and finitely branching tree domain. The full construction is an extension of the vertical collapsing construction, and proceeds in stages, considering at each Stage i the nodes at depth i. At each stage, we are initially concerned with the branching factor. The construction we now give takes a node and produces a model in which that node has only finitely many children (while preserving the invariants). Applying this to all nodes at depth $i-1$ results in a model with finitely many nodes at depth i. We can then proceed with Stage i of the vertical collapsing construction.

Let A be a model of $\mathcal{A}_{\mathbf{Fin}}$. Let a node $a \in |A|$. We construct a model A' such that $a \in |A'| \subseteq |A|$, the number of children of a in A' is finite, $A' \equiv_n A$, and, for all nodes $a' \in |A'|$, $A'{\downarrow}_{a'} \equiv_n A{\downarrow}_{a'}$.

The construction proceeds in two stages. First we identify a sequence of pairs of the children of a such that the pairs meet the hypothesis of Lemma 37 and all but finitely many of the children of a fall between pairs. In the second phase, we horizontally collapse the model at these pairs, thereby deleting all but finitely many of the children.

Phase 1
We construct a sequence $\langle a_0, b_0\rangle, \langle a_1, b_1\rangle \ldots$ of pairs of children of a in A as follows. If a has any children then, as A is a model of $\mathcal{A}$, a has an unique leftmost child. Let a_0 be the leftmost child of a. Suppose we have a_i. By Corollary 14 the $n, 2$-type of $\langle a, a_i\rangle$ is characterized by a formula $\chi^n_{A,\langle a,a_i\rangle}(x, y)$. Let $\chi'_i(y)$ be

$$(\exists x)[x \triangleleft y \wedge \chi^n_{A,\langle a,a_i\rangle}(x, y)].$$

Since a_i satisfies χ'_i in A, by **Fin-B** there is some maximal child of a, possibly a_i itself, that satisfies χ'_i in A. Let this node be b_i. If b_i has any right-siblings then b_i has an unique immediate right-sibling (again, because A is a model of $\mathcal{A}$). Let a_{i+1} be the immediate right sibling of b_i, if any.

Because each of the b_i is chosen to be the maximal child of its $n, 2$-type (with a), there is no right-sibling of b_i that has the same $n, 2$-type as any b_j for $j \leq i$. By Corollary 15, there are but finitely many distinct $n, 2$-types realized in A. Thus there is some i less than or equal to that limit for which b_i has no right siblings. At that point, this phase of the construction terminates.

Phase 2
We have from the first phase a finite sequence of pairs: $\langle a_0, b_0\rangle, \ldots, \langle a_l, b_l\rangle$. We construct a sequence of models by applying horizontal collapsings at the pairs in this sequence *in reverse.* Thus, this sequence of models can be denoted by

$$A = A_{l+1}, A_l, \ldots, A_0 = A',$$

where $A_i = \mathrm{HC}(A_{i+1}, a_i, b_i)$. Clearly, $|A_i| \subseteq |A_i + 1|$ for all $i \leq l$, and, thus, $|A'| \subseteq |A|$.

Note that each pair $\langle a_i, b_i\rangle$ $(0 \leq i \leq l)$ in the sequence of Phase 1 satisfies the conditions of the hypothesis of Lemmas 39 and 37. By considering this sequence in reverse, if we collapse at $\langle a_i, b_i\rangle$ we can be guaranteed these conditions are still satisfied for the pairs that will be collapsed later. That is, by Lemma 39, we know that collapsing at a_i and b_i does not affect the $n, 2$-type with a of a_j or b_j for any $j < i$. Thus, for $j < i$, the $n, 2$-type with a of a_j and b_j will still be equal after collapsing of $\langle a_i, b_i\rangle$. The hypothesis of this lemma, then, will always hold for all $i \leq l$. Now similarly, by Corollary 38, we have $A_i \equiv_n A_{i+1}$, and by transitivity of equivalence $A' \equiv_n A$. By Lemma 37, the construction preserves the types of the subtrees rooted at nodes in A'. Finally, the children of a in A' are exactly the b_i, and there are but $l + 1$ of these.

Given a and A, we will say Finite-branching(A, a) to denote the A' obtained by this construction.

4.2.1. *The Combined Construction*

We now can establish that for every model of $\mathcal{A}_{\mathbf{Fin}}$ there is an n-equivalent finite tree, for every n. Previously we have seen how we could use vertical collapsing to construct finite-depth trees. In that construction, given in Section 4.1, at the i^{th} stage, we considered nodes at depth i (where the root was at depth 0). If a was such a node, we found a maximal node b such that the subtrees rooted at these two nodes were n-equivalent. At the next stage, the children of b were considered. That there were only finitely many children followed because we were concerned with models of $\mathcal{A}_{\mathbf{BBn}}$. Now, a model of $\mathcal{A}_{\mathbf{Fin}}$ could have nodes with possibly infinitely many children. However, we can use the horizontal collapsing construction to ensure that, before we consider the next depth, there will only be finitely many nodes at that depth.

Let A be a model of $\mathcal{A}_{\mathbf{Fin}}$. Again we construct a sequence of models that are n-equivalent to A, ending in a finite-tree. Now, however, we alternate between collapsing horizontally and vertically and construct a sequence

$$A = A_0 = A'_0, A_1, A'_1, \ldots, A_l, A'_l.$$

At Stage 0, we consider the root, a_0 of A_0. As a_0 has no siblings, no horizontal collapsing is necessary. Let b_0 be the maximal node dominated by a_0 such that $(A_0{\downarrow}_{a_0}, a_0) \equiv_n (A_0{\downarrow}_{b_0}, b_0)$. As A_0 is a model of **Fin-D**, such a node exists. Let $A_1 = \mathrm{VC}(A_0, a_0, b_0)$.

Stage $i \geq 1$

By construction there will be finitely many nodes at depth $i-1$ in A_i. Let these nodes be $a_{\langle i-1,1\rangle}, \ldots, a_{\langle i-1,m_{i-1}\rangle}$. We construct a sequence of models $A_i = A_{\langle i,1\rangle}, \ldots A_{\langle i,m_{i-1}\rangle} = A'_i$ by letting $A_{\langle i,k+1\rangle} = \text{Finite-branching}(A_{\langle i,k\rangle}, a_{\langle i-1,k\rangle})$. This means that in A'_i, all nodes at depth $i-1$ have finite number of children. Now we can consider these children, which are at depth i, and perform vertical collapsing as indicated in the construction in Section 4.1. That is, in A'_i, the nodes at depth i can be denoted as $a_{i,j}$ ($0 \leq j \leq m_i$, for some $m_i \in \mathbb{N}$). For each $a_{i,j}$ ($0 \leq j \leq m_i$), we find a maximal $b_{i,j}$ such that $(A'_{i-1}{\downarrow}_{a_{i,j}}, a_{i,j}) \equiv_n (A'_{i-1}{\downarrow}_{b_{i,j}}, b_{i,j})$ as before. Let A_i be the vertical collapse of A_{i-1} at each of the $a_{i,j}, b_{i,j}$ in turn.

LEMMA 40. *The construction just outlined terminates in finitely many steps, and results in a finite tree that is n-equivalent to A.*

This follows from the equivalent arguments for the individual components of the construction. Finally, this establishes our main result, that $\mathcal{A}_{\mathbf{Fin}}$ implies exactly the first-order theory of finite trees.

LEMMA 41. *For any sentence ψ in L, if ψ is consistent with $\mathcal{A}$, all instances of* **Fin-B**, *and all instances of* **Fin-D**, *then ψ is satisfied by a finite tree.*

THEOREM 42. *The first-order consequences of $\mathcal{A}_{\mathbf{Fin}}$ are exactly the first-order theory of finite trees.*

5. CONCLUDING REMARKS

There has been a growing body of work in linguistics involving formal arguments about the structure of trees. Our results address the foundations of this work. We have provided a set of first-order axioms $\mathcal{A}$ that capture the properties of trees that form the basis for these arguments. We have shown, though, that these axioms do not suffice to define the class of structures that are trees, and that, in fact, no set of first-order sentences can do so. Nonetheless, by adding the schema **Fin-D** and either the axiom **BBn** for some $n \in \mathbb{N}$ or the schema **Fin-B** to these basic axioms, we obtain a recursive set of first-order axioms that imply exactly the first-order theory of finite trees with bounded branching or finite trees with arbitrary (finite) branching, respectively. Moreover, we show that adding these schemas to $\mathcal{A}$ is equivalent to enhancing one's deductive mechanism with inferences based on induction on the depth of nodes and on the number of siblings preceding nodes (coupled with inferences from the fact that every branch and every set of children is bounded). Such inferences are typical of formal arguments about the structure of trees. Our result then, confirms that such arguments are, at least in principle, capable of deriving every first-order property of trees. This is the case even when the inductions are applied only to properties that are expressible in our first-order language.

It should be noted that our structures model only the skeletons of trees. In linguistic usage, the nodes of the trees are decorated with labels and features indicating various categories and the roles of the nodes in the syntactic structure. As long as these decorations can be resolved into a finite set of atomic features, that is, as long as they ultimately distinguish finitely many subsets of the nodes in the trees, we can capture them as monadic second-order predicates. As we noted earlier in passing, Doets's results (Doets, 1989) actually concern first-order axiomatizations of monadic Π^1_1-theories, the universal fragment of monadic second-order theories. Following his approach, we can expand our language to include finitely many monadic predicate symbols, and extend our schema to include instances for every formula in the expanded language. This does not alter our proofs. As there are only finitely many additional predicates the number of n, k-types is only multiplied by some finite factor (which depends on n and k as well as the number of predicates). These types are still characterized by individual formulae and the proofs go through exactly as before. We have, then, a recursive set of axioms that capture the monadic Π^1_1-theory of finite trees, that is, the universal fragment of the theory of finite trees labeled with atomic features. Furthermore, deduction from these axioms is equivalent to deduction from the basic set $\mathcal{A}$ enhanced with induction, as above, but applied here to every property that is expressible in the first-order language using finitely many monadic parameters.

It is easy to show that this theory can be embedded in SnS—the monadic second-order theory of multiple successor functions (Rogers, 1994). In a celebrated result, Rabin showed that SnS is decidable (Rabin, 1969). It follows that the theories we axiomatize are also decidable.[10] Thus not only are all monadic Π^1_1-properties of finite trees derivable from these axioms, the question of whether a given sentence expresses such a property, or equivalently, if a given sentence is satisfied by any finite tree, can be resolved algorithmically.

Thus far these results argue for the strength of these axioms in establishing linguistic results about the structure of trees. But the fact that the theory is embeddable in SnS also gives us an upper bound on the kinds of properties that can be expressed within the theory and, hence, an upper bound on the kinds of properties that can be derived from these axioms. It has been shown, originally by Doner (1970), that the class of sets of finite trees that are definable in SnS is exactly the class of *recognizable sets.* The recognizable sets are essentially the class of sets of derivation trees that can be generated by Context-Free Grammars.[11] Thus every string language that is the yield of a set of finite trees that is definable in our language (augmented with finitely many monadic second-order parameters) is strongly Context-Free. Furthermore, this bound is tight since it is easy to construct, given any CFG G, a sentence ϕ_G in L (augmented with parameters for the terminal and non-terminal symbols of G) such that consequences of $\mathcal{A}_{\mathbf{Fin}} \cup \{\phi_G\}$ are exactly the sentences in the augmented language that are true in every tree generated by G. Consequently, there is no monadic Π^1_1-property of trees,[12] and thus no property that can be derived from these axioms, that cannot be enforced by a Context-Free Grammar and *vice versa.* To define sets of trees that embody properties that are beyond the power of CFGs, or, equivalently, to establish results about such properties, one must either resort to extra-logical mechanisms or expand the language, by including, for instance, non-monadic predicates (a single arbitrary binary relation suffices), or by employing non-atomic labels (as in Blackburn *et al.*).

Acknowledgements

This work has benefited greatly from discussions with Eric Rosen, Scott Weinstein, and Steven Lindell, and especially from the detailed commentary of an anonymous reviewer.

[10] Such decidability does not follow here, as it often does, from the existence of a recursive axiomatization because the theory is not *complete*; the fact that a sentence is not in the theory does not imply that its negation is.

[11] These are termed *local sets.* Technically every recognizable set is the projection of a local set.

[12] In fact, no monadic second-order property of trees whatsoever.

References

Patrick Blackburn and Wilfried Meyer-Viol. Linguistics, logic, and finite trees. *Bulletin of the IGPL*, 2:3–29, 1994.

Patrick Blackburn, Claire Gardent, and Wilfried Meyer-Viol. Talking about trees. In *Proceedings of the 6th Conference of the EACL*, pages 21–29. European Chapter of the Association for Computational Linguistics, 1993.

Thomas Longacre Cornell. *Description Theory, Licensing Theory, and Principle-Based Grammars and Parsers.* PhD thesis, University of California Los Angeles, 1992.

Bruno Courcelle. Fundamental properties of infinite trees. *Theoretical Computer Science*, 25:95–169, 1983.

Kees Doets. Monadic Π^1_1-theories of Π^1_1-properties. *Notre Dame Journal of Formal Logic*, 30(2):224–240, 1989.

John Doner. Tree acceptors and some of their applications. *Journal of Computer and System Sciences*, 4:406–451, 1970.

H.-D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic.* Springer-Verlag, New York, 1984.

G. Gazdar, E. Klein, G. Pullum, and I. Sag. *Generalized Phrase Structure Grammar.* Basil Blackwell, 1985.

James Henderson. Structure Unification Grammar: A unifying framework for investigating natural language. Master's thesis, University of Pennsylvania, Phila., PA, December 1990.

Mark Johnson. The use of knowledge of language. *Journal of Psycholinguistic Research*, 18(1):105–128, 1989.

Richard S. Kayne. Unambiguous paths. In R. May and J. Koster, editors, *Levels of Syntactic Representation*, pages 143–183. Foris, Dordrecht, 1981.

Richard S. Kayne. *The Antisymmetry of Syntax.* MIT Press, Cambridge, MA, 1994.

Marcus Kracht. Mathematical aspects of command relations. In *Proceedings of the 6th Conference of the EACL*. Euorpean Chapter of the Association for Computational Linguistics, 1993.

Michael J. Maher. Complete axiomatizations of the algebras of finite, rational and infinite trees. In *Proceedings of the 3rd Annual Symposium on Logic in Computer Science*, pages 348–457, Edinburgh, Scotland, July 1988.

Mitchell P. Marcus, Donald Hindle, and Margaret M. Fleck. D-theory: Talking about talking about trees. In *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, 1983.

Barbara Partee, Alice ter Meulen, and Robert Wall. *Mathematical Methods in Linguistics*, volume 20 of *Studies in Linguistics and Philosophy.* Kluwer, Dordrecht, 1990.

Michael O. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the American Mathematical Society*, 141:1–35, July 1969.

James Rogers and K. Vijay-Shanker. Obtaining trees from their descriptions: An application to Tree-Adjoining Grammars. *Computational Intelligence*, 10:401–421, 1994.

James Rogers. *Studies in the Logic of Trees with Applications to Grammar Formalisms.* Ph.D. dissertation, Univ. of Delaware, 1994.

Dirk Siefkes. An axiom system for the weak monadic second order theory of two successors. *Israel Journal of Mathematics*, 30(3):264–284, 1978.

Edward P. Stabler, Jr. *The Logical Approach to Syntax.* Bradford, 1992.

K. Vijay-Shanker. Using descriptions of trees in a Tree-Adjoining Grammar. *Computational Linguistics*, 18(4):481–517, 1992.

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

-Bibliothek, Information
und Dokumentation (BID)-
PF 2080
67608 Kaiserslautern
FRG

Telefon (0631) 205-3506
Telefax (0631) 205-3210
e-mail
dfkibib@dfki.uni-kl.de
WWW
http://www.dfki.uni-sb.de/dfkibib

Veröffentlichungen des DFKI

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse oder (so sie als per ftp erhaeltlich angemerkt sind) per anonymous ftp von ftp.dfki.uni-kl.de (131.246.241.100) im Verzeichnis pub/Publications bezogen werden. Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all published papers so far are obtainable from the above address or (if they are marked as obtainable by ftp) by anonymous ftp from ftp.dfki.uni-kl.de (131.246.241.100) in the directory pub/Publications.
The reports are distributed free of charge except where otherwise noted.

DFKI Research Reports

1995

RR-95-02
Majdi Ben Hadj Ali, Frank Fein, Frank Hoenes, Thorsten Jaeger, Achim Weigel
Document Analysis at DFKI
Part 1: Image Analysis and Text Recognition
69 pages

1994

RR-94-39
Hans-Ulrich Krieger
Typed Feature Formalisms as a Common Basis for Linguistic Specification.
21 pages

RR-94-38
Hans Uszkoreit, Rolf Backofen, Stephan Busemann, Abdel Kader Diagne, Elizabeth A. Hinkelman, Walter Kasper, Bernd Kiefer, Hans-Ulrich Krieger, Klaus Netter, Günter Neumann, Stephan Oepen, Stephen P. Spackman.
DISCO-An HPSG-based NLP System and its Application for Appointment Scheduling.
13 pages

RR-94-37
Hans-Ulrich Krieger, Ulrich Schaefer.
TDL - A Type Description Language for HPSG, Part 1: Overview.
54 pages

RR-94-36
Manfred Meyer
Issues in Concurrent Knowledge Engineering. Knowledge Base and Knowledge Share Evolution.
17 pages

RR-94-35
Rolf Backofen
A Complete Axiomatization of a Theory with Feature and Arity Constraints
49 pages

RR-94-34
Stephan Busemann, Stephan Oepen, Elizabeth A. Hinkelman, Günter Neumann, Hans Uszkoreit
COSMA - Multi-Participant NL Interaction for Appointment Scheduling
80 pages

RR-94-33
Franz Baader, Armin Laux
Terminological Logics with Modal Operators
29 pages

RR-94-31
Otto Kühn, Volker Becker, Georg Lohse, Philipp Neumann
Integrated Knowledge Utilization and Evolution for the Conservation of Corporate Know-How
17 pages

RR-94-23
Gert Smolka
The Definition of Kernel Oz
53 pages

RR-94-20
Christian Schulte, Gert Smolka, Jörg Würtz
Encapsulated Search and Constraint Programming in Oz
21 pages

RR-94-18
Rolf Backofen, Ralf Treinen
How to Win a Game with Features
18 pages

RR-94-17
Georg Struth
Philosophical Logics—A Survey and a Bibliography
58 pages

RR-94-16
Gert Smolka
A Foundation for Higher-order Concurrent Constraint Programming
26 pages

RR-94-15
Winfried H. Graf, Stefan Neurohr
Using Graphical Style and Visibility Constraints for a Meaningful Layout in Visual Programming Interfaces
20 pages

RR-94-14
Harold Boley, Ulrich Buhrmann, Christof Kremer
Towards a Sharable Knowledge Base on Recyclable Plastics
14 pages

RR-94-13
Jana Koehler
Planning from Second Principles—A Logic-based Approach
49 pages

RR-94-12
Hubert Comon, Ralf Treinen
Ordering Constraints on Trees
34 pages

RR-94-11
Knut Hinkelmann
A Consequence Finding Approach for Feature Recognition in CAPP
18 pages

RR-94-10
Knut Hinkelmann, Helge Hintze
Computing Cost Estimates for Proof Strategies
22 pages

RR-94-08
Otto Kühn, Björn Höfling
Conserving Corporate Knowledge for Crankshaft Design
17 pages

RR-94-07
Harold Boley
Finite Domains and Exclusions as First-Class Citizens
25 pages

RR-94-06
Dietmar Dengler
An Adaptive Deductive Planning System
17 pages

RR-94-05
Franz Schmalhofer, J. Stuart Aitken, Lyle E. Bourne jr.
Beyond the Knowledge Level: Descriptions of Rational Behavior for Sharing and Reuse
81 pages

RR-94-03
Gert Smolka
A Calculus for Higher-Order Concurrent Constraint Programming with Deep Guards
34 pages

RR-94-02
Elisabeth André, Thomas Rist
Von Textgeneratoren zu Intellimedia-Präsentationssystemen
22 Seiten

RR-94-01
Elisabeth André, Thomas Rist
Multimedia Presentations: The Support of Passive and Active Viewing
15 pages

1993

RR-93-48
Franz Baader, Martin Buchheit, Bernhard Hollunder
Cardinality Restrictions on Concepts
20 pages

RR-93-46
Philipp Hanschke
A Declarative Integration of Terminological, Constraint-based, Data-driven, and Goal-directed Reasoning
81 pages

RR-93-45
Rainer Hoch
On Virtual Partitioning of Large Dictionaries for Contextual Post-Processing to Improve Character Recognition
21 pages

RR-93-44
Martin Buchheit, Manfred A. Jeusfeld, Werner Nutt, Martin Staudt
Subsumption between Queries to Object-Oriented Databases
36 pages

RR-93-43
M. Bauer, G. Paul
Logic-based Plan Recognition for Intelligent Help Systems
15 pages

RR-93-42
Hubert Comon, Ralf Treinen
The First-Order Theory of Lexicographic Path Orderings is Undecidable
9 pages

RR-93-41
Winfried H. Graf
LAYLAB: A Constraint-Based Layout Manager for Multimedia Presentations
9 pages

RR-93-40
Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt, Andrea Schaerf
Queries, Rules and Definitions as Epistemic Statements in Concept Languages
23 pages

RR-93-38
Stephan Baumann
Document Recognition of Printed Scores and Transformation into MIDI
24 pages

RR-93-36
Michael M. Richter, Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Gabriele Schmidt
Von IDA bis IMCOD: Expertensysteme im CIM-Umfeld
13 Seiten

RR-93-35
Harold Boley, François Bry, Ulrich Geske (Eds.)
Neuere Entwicklungen der deklarativen KI-Programmierung — *Proceedings*
150 Seiten

Note: This document is available for a nominal charge of 25 DM (or 15 US-$).

RR-93-34
Wolfgang Wahlster
Verbmobil Translation of Face-To-Face Dialogs
10 pages

RR-93-33
Bernhard Nebel, Jana Koehler
Plan Reuse versus Plan Generation: A Theoretical and Empirical Analysis
33 pages

RR-93-32
David R. Traum, Elizabeth A. Hinkelman
Conversation Acts in Task-Oriented Spoken Dialogue
28 pages

RR-93-31
Elizabeth A. Hinkelman, Stephen P. Spackman
Abductive Speech Act Recognition, Corporate Agents and the COSMA System
34 pages

RR-93-30
Stephen P. Spackman, Elizabeth A. Hinkelman
Corporate Agents
14 pages

RR-93-29
Armin Laux
Representing Belief in Multi-Agent Worlds via Terminological Logics
35 pages

RR-93-28
Hans-Ulrich Krieger, John Nerbonne, Hannes Pirker
Feature-Based Allomorphy
8 pages

RR-93-27
Hans-Ulrich Krieger
Derivation Without Lexical Rules
33 pages

RR-93-26
Jörg P. Müller, Markus Pischel
The Agent Architecture InteRRaP: Concept and Application
99 pages

RR-93-25
Klaus Fischer, Norbert Kuhn
A DAI Approach to Modeling the Transportation Domain
93 pages

RR-93-24
Rainer Hoch, Andreas Dengel
Document Highlighting — Message Classification in Printed Business Letters
17 pages

RR-93-23
Andreas Dengel, Ottmar Lutzy
Comparative Study of Connectionist Simulators
20 pages

RR-93-22
Manfred Meyer, Jörg Müller
Weak Looking-Ahead and its Application in Computer-Aided Process Planning
17 pages

RR-93-20
Franz Baader, Bernhard Hollunder
Embedding Defaults into Terminological Knowledge Representation Formalisms
34 pages

RR-93-18
Klaus Schild
Terminological Cycles and the Propositional μ-Calculus
32 pages

RR-93-17
Rolf Backofen
Regular Path Expressions in Feature Logic
37 pages

RR-93-16
Gert Smolka, Martin Henz, Jörg Würtz
Object-Oriented Concurrent Constraint Programming in Oz
17 pages

RR-93-15
Frank Berger, Thomas Fehrle, Kristof Klöckner, Volker Schölles, Markus A. Thies, Wolfgang Wahlster
PLUS - Plan-based User Support Final Project Report
33 pages

RR-93-14
Joachim Niehren, Andreas Podelski, Ralf Treinen
Equational and Membership Constraints for Infinite Trees
33 pages

RR-93-13
Franz Baader, Karl Schlechta
A Semantics for Open Normal Defaults via a Modified Preferential Approach
25 pages

RR-93-12
Pierre Sablayrolles
A Two-Level Semantics for French Expressions of Motion
51 pages

RR-93-11
Bernhard Nebel, Hans-Jürgen Bürckert
Reasoning about Temporal Relations: A Maximal Tractable Subclass of Allen's Interval Algebra
28 pages

RR-93-10
Martin Buchheit, Francesco M. Donini, Andrea Schaerf
Decidable Reasoning in Terminological Knowledge Representation Systems
35 pages

RR-93-09
Philipp Hanschke, Jörg Würtz
Satisfiability of the Smallest Binary Program
8 pages

RR-93-08
Harold Boley, Philipp Hanschke, Knut Hinkelmann, Manfred Meyer
CoLab: A Hybrid Knowledge Representation and Compilation Laboratory
64 pages

RR-93-07
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux
Concept Logics with Function Symbols
36 pages

RR-93-06
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux
On Skolemization in Constrained Logics
40 pages

RR-93-05
Franz Baader, Klaus Schulz
Combination Techniques and Decision Problems for Disunification
29 pages

RR-93-04
Christoph Klauck, Johannes Schwagereit
GGD: Graph Grammar Developer for features in CAD/CAM
13 pages

RR-93-03
Franz Baader, Berhard Hollunder, Bernhard Nebel, Hans-Jürgen Profitlich, Enrico Franconi
An Empirical Analysis of Optimization Techniques for Terminological Representation Systems
28 pages

RR-93-02
Wolfgang Wahlster, Elisabeth André, Wolfgang Finkler, Hans-Jürgen Profitlich, Thomas Rist
Plan-based Integration of Natural Language and Graphics Generation
50 pages

RR-93-01
Bernhard Hollunder
An Alternative Proof Method for Possibilistic Logic and its Application to Terminological Logics
25 pages

DFKI Technical Memos

1995

TM-95-01
Martin Buchheit, Rüdiger Klein, Werner Nutt
Constructive Problem Solving: A Model Construction Approach towards Configuration
34 pages

1994

TM-94-04
Cornelia Fischer
PAntUDE - An Anti-Unification Algorithm for Expressing Refined Generalizations
22 pages

TM-94-03
Victoria Hall
Uncertainty-Valued Horn Clauses
31 pages

TM-94-02
Rainer Bleisinger, Berthold Kröll
Representation of Non-Convex Time Intervals and Propagation of Non-Convex Relations
11 pages

TM-94-01
Rainer Bleisinger, Klaus-Peter Gores
Text Skimming as a Part in Paper Document Understanding
14 pages

1993

TM-93-05
Michael Sintek
Indexing PROLOG Procedures into DAGs by Heuristic Classification
64 pages

TM-93-04
Hans-Günther Hein
Propagation Techniques in WAM-based Architectures — The FIDO-III Approach
105 pages

TM-93-03
Harold Boley, Ulrich Buhrmann, Christof Kremer
Konzeption einer deklarativen Wissensbasis über recyclingrelevante Materialien
11 pages

TM-93-02
Pierre Sablayrolles, Achim Schupeta
Conflict Resolving Negotiation for COoperative Schedule Management Agents (COSMA)
21 pages

TM-93-01
Otto Kühn, Andreas Birk
Reconstructive Integrated Explanation of Lathe Production Plans
20 pages

DFKI Documents

1995

D-95-02
Andreas Butz
BETTY
Ein System zur Planung und Generierung informativer Animationssequenzen
95 Seiten

D-95-01
Susanne Biundo, Wolfgang Tank (Hrsg.)
Beitraege zum Workshop „Planen und Konfigurieren", Februar 1995
169 Seiten

Note: This document is available for a nominal charge of 25 DM (or 15 US-$).

1994

D-94-15
Stephan Oepen
German Nominal Syntax in HPSG
— On Syntactic Categories and Syntagmatic Relations
—
80 pages

D-94-14
Hans-Ulrich Krieger, Ulrich Schaefer.
TDL - A Type Description Language for HPSG, Part 2: User Guide.
72 pages

D-94-12
Arthur Sehn, Serge Autexier (Hrsg.)
Proceedings des Studentenprogramms der 18. Deutschen Jahrestagung für Künstliche Intelligenz KI-94
69 Seiten

D-94-11
F. Baader, M. Buchheit, M. A. Jeusfeld, W. Nutt (Eds.)
Working Notes of the KI'94 Workshop: KRDB'94 - Reasoning about Structured Objects: Knowledge Representation Meets Databases
65 pages

Note: This document is no longer available in printed form.

D-94-10
F. Baader, M. Lenzerini, W. Nutt, P. F. Patel-Schneider (Eds.)
Working Notes of the 1994 International Workshop on Description Logics
118 pages

Note: This document is available for a nominal charge of 25 DM (or 15 US-$).

D-94-09
Technical Staff
DFKI Wissenschaftlich-Technischer Jahresbericht 1993
145 Seiten

D-94-08
Harald Feibel
IGLOO 1.0 - Eine grafikunterstützte Beweisentwicklungsumgebung
58 Seiten

D-94-07
Claudia Wenzel, Rainer Hoch
Eine Übersicht über Information Retrieval (IR) und NLP-Verfahren zur Klassifikation von Texten
25 Seiten

D-94-06
Ulrich Buhrmann
Erstellung einer deklarativen Wissensbasis über recyclingrelevante Materialien
117 Seiten

D-94-04
Franz Schmalhofer, Ludger van Elst
Entwicklung von Expertensystemen: Prototypen, Tiefenmodellierung und kooperative Wissensevolution
22 Seiten

D-94-03
Franz Schmalhofer
Maschinelles Lernen: Eine kognitionswissenschaftliche Betrachtung
54 Seiten

Note: This document is no longer available in printed form.

D-94-02
Markus Steffens
Wissenserhebung und Analyse zum Entwicklungsprozeß eines Druckbehälters aus Faserverbundstoff
90 pages

D-94-01
Josua Boon (Ed.)
DFKI-Publications: The First Four Years
1990 - 1993
75 pages

1993

D-93-27
Rolf Backofen, Hans-Ulrich Krieger, Stephen P. Spackman, Hans Uszkoreit (Eds.)
Report of the EAGLES Workshop on Implemented Formalisms at DFKI, Saarbrücken
110 pages

D-93-26
Frank Peters
Unterstützung des Experten bei der Formalisierung von Textwissen INFOCOM - Eine interaktive Formalisierungskomponente
58 Seiten

D-93-25
Hans-Jürgen Bürckert, Werner Nutt (Eds.)
Modeling Epistemic Propositions
118 pages

Note: This document is available for a nominal charge of 25 DM (or 15 US-$).

D-93-24
Brigitte Krenn, Martin Volk
DiTo-Datenbank: Datendokumentation zu Funktionsverbgefügen und Relativsätzen
66 Seiten

D-93-22
Andreas Abecker
Implementierung graphischer Benutzungsoberflächen mit Tcl/Tk und Common Lisp
44 Seiten

Note: This document is no longer available in printed form.

D-93-21
Dennis Drollinger
Intelligentes Backtracking in Inferenzsystemen am Beispiel Terminologischer Logiken
53 Seiten

D-93-20
Bernhard Herbig
Eine homogene Implementierungsebene für einen hybriden Wissensrepräsentationsformalismus
97 Seiten

D-93-16
Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Gabriele Schmidt
Design & KI
74 Seiten

D-93-15
Robert Laux
Untersuchung maschineller Lernverfahren und heuristischer Methoden im Hinblick auf deren Kombination zur Unterstützung eines Chart-Parsers
86 Seiten

D-93-14
Manfred Meyer (Ed.)
Constraint Processing – Proceedings of the International Workshop at CSAM'93, St.Petersburg, July 20-21, 1993
264 pages

Note: This document is available for a nominal charge of 25 DM (or 15 US-$).

D-93-12
Harold Boley, Klaus Elsbernd, Michael Herfert, Michael Sintek, Werner Stein
RELFUN Guide: Programming with Relations and Functions Made Easy
86 pages

D-93-11
Knut Hinkelmann, Armin Laux (Eds.)
DFKI Workshop on Knowledge Representation Techniques — Proceedings
88 pages

Note: This document is no longer available in printed form.

D-93-10
Elizabeth Hinkelman, Markus Vonerden, Christoph Jung
Natural Language Software Registry (Second Edition)
174 pages

D-93-09
Hans-Ulrich Krieger, Ulrich Schäfer
$\mathcal{TDL}$*ExtraLight* User's Guide
35 pages

D-93-08
Thomas Kieninger, Rainer Hoch
Ein Generator mit Anfragesystem für strukturierte Wörterbücher zur Unterstützung von Texterkennung und Textanalyse
125 Seiten

D-93-07
Klaus-Peter Gores, Rainer Bleisinger
Ein erwartungsgesteuerter Koordinator zur partiellen Textanalyse
53 Seiten

D-93-06
Jürgen Müller (Hrsg.)
Beiträge zum Gründungsworkshop der Fachgruppe Verteilte Künstliche Intelligenz, Saarbrücken, 29. - 30. April 1993
235 Seiten

Note: This document is available for a nominal charge of 25 DM (or 15 US-$).

D-93-05
Elisabeth André, Winfried Graf, Jochen Heinsohn, Bernhard Nebel, Hans-Jürgen Profitlich, Thomas Rist, Wolfgang Wahlster
PPP: Personalized Plan-Based Presenter
70 pages

D-93-04
Technical Staff
DFKI Wissenschaftlich-Technischer Jahresbericht 1992
194 Seiten

D-93-03
Stephan Busemann, Karin Harbusch(Eds.)
DFKI Workshop on Natural Language Systems: Reusability and Modularity - Proceedings
74 pages

D-93-02
Gabriele Schmidt, Frank Peters, Gernod Laufkötter
User Manual of COKAM+
23 pages

D-93-01
Philipp Hanschke, Thom Frühwirth
Terminological Reasoning with Constraint Handling Rules
12 pages

A First-Order Axiomatization of the Theory of Finite Trees

Rolf Backofen, James Rogers, K. Vijay-Shanker

RR-95-05

Research Report