THE JOURNAL OF SYMBOLIC LOGIC
Volume 48, Number 4, Dec. 1983

RABIN'S UNIFORMIZATION PROBLEM[1]

YURI GUREVICH[2] AND SAHARON SHELAH

Abstract. The set of all words in the alphabet $\{l, r\}$ forms the *full binary tree* T. If $x \in T$ then xl and xr are the *left* and the *right successors* of x respectively. We consider the monadic second-order language of the full binary tree with the two successor relations. This language allows quantification over elements of T and over arbitrary subsets of T. We prove that there is no monadic second-order formula $\phi^*(X, y)$ such that for every nonempty subset X of T there is a unique $y \in X$ that satisfies $\phi^*(X, y)$ in T.

INTRODUCTION

The *uniformization problem* for a theory TH in a formalized language L can be formulated as follows. Suppose TH $\vdash \forall\tilde{u}\exists\tilde{v}\phi(\tilde{u}, \tilde{v})$ where ϕ is an L-formula and $\tilde{u}$, $\tilde{v}$ are tuples of variables. Is there another L-formula $\phi^*(\tilde{u}, \tilde{v})$ such that

$$\mathrm{TH} \vdash \forall\tilde{u}\forall\tilde{v}(\phi^*(\tilde{u}, \tilde{v}) \to \phi(\tilde{u}, \tilde{v})) \quad \text{and} \quad \mathrm{TH} \vdash \forall\tilde{u}\exists!\,\tilde{v}\phi^*(\tilde{u}, \tilde{v})?$$

Here $\exists!$ means "there is a unique".

Rabin's Uniformization Problem is the uniformization problem for the monadic second-order theory of the full binary tree. Let us recall necessary definitions and survey very briefly the history of the problem.

The monadic second-order logic is the fragment of the full second-order logic that allows quantification over elements and over monadic predicates only. One way to define the monadic version of the first-order language L is to augment L by a list of quantifiable set variables and by new atomic formulas $t \in X$ where t is a first-order term and X is a set variable. Suppose that M is a structure for L. The monadic second-order theory of M is the theory of M in the extended language when the set variables range over all subsets of M and $\in$ is the containment relation.

The monadic second-order theory of the structure (ω, Successor) is known as S1S. Here ω is the set of natural numbers, Successor is the usual successor operation, and S1S abbreviates "the second-order theory of one successor". The decision problem for S1S was solved positively by Büchi [1962]. The uniformization problem for S1S was solved positively by Büchi and Landweber [1969]. Let $\phi(\tilde{u}, \tilde{v})$ be a formula in the language of S1S. We can view each $\tilde{u}$ and $\tilde{v}$ as an ω-sequence of letters in a certain finite alphabet. Büchi and Landweber proved that if

Received January 11, 1982.

[1]The work was done in principal during the 1980–81 academic year when both authors were fellows in the Institute for Advanced Studies of the Hebrew University in Jerusalem.

[2]Address for correspondence: Department of Computer and Communication Sciences, University of Michigan, Ann Arbor, MI 48109, USA.

0022-4812/83/4804-0017/$02.50

S1S $\vdash \forall \tilde{u} \exists \tilde{v} \phi(\tilde{u}, \tilde{v})$ then there is a finite automaton that outputs an appropriate ω-sequence $\tilde{v}$ when it reads the given ω-sequence $\tilde{u}$. The desired uniformizing formula ϕ^* describes that finite automaton.

The *full binary tree* T is here the set of all words in the alphabet $\{l, r\}$. The monadic second-order theory of T with two successor operations $\text{Successor}_l(x) = xl$ and $\text{Successor}_r(x) = xr$ is known as S2S, which is an abbreviation for "the second-order theory of two successors". Rabin [1969] solved positively the decision problem for S2S. To simplify the notation we describe here only a partial case of a result from Rabin [1972]. Let $\phi(X)$ be a formula in the language of S2S where X is a set variable. If S2S $\vdash \exists X \phi(X)$, then there is a regular (i.e., recognizable by a finite automaton) subset X of the full binary tree such that $\phi(X)$ holds in T.

The quoted results of Büchi, Büchi and Landweber, and Rabin can be seen as a hint for a positive solution for the uniformization problem for S2S. This paper gives, however, a negative solution for that uniformization problem. More specifically, we prove the following: Let $\phi(X, y)$ be a formula in the language of S2S saying that if X is nonempty then $y \in X$. Clearly S2S $\vdash \forall X \exists y \phi(X, y)$. We prove that there is no formula $\phi^*(X, y)$ in the language of S2S such that for every $X \subseteq T$ there is a unique $y \in T$ satisfying $\phi(X, y)$ in T.

The proof is model-theoretic in its nature. The main tool is the Composition Theorem for trees proved in Chapter 2. Another tool is the Addition Theorem for intervals of trees proved in Chapter 3. Rabin's Uniformization Problem is solved in Chapter 5 with use of forcing. The reader can go on straightway and read §5.1 in order to get some idea of how the proof goes.

Finally, let us note that we allow only set variables in our monadic second-order languages. The original structure is coded on singleton sets. This way we transform a given monadic second-order theory into the first-order theory of certain special structures.

CHAPTER 2. GRAFTING

The main goal of this chapter is the Composition Theorem for trees. In §2.1 we recall the notion of n-theory due to Läuchli [1968] and its important generalization due to Shelah [1975]. The Composition Theorem for trees is proved in §2.3. Its proof follows the routine developed by Shelah [1975] and Gurevich [1979]. For the reader's convenience we make this exposition self-contained. In §2.4 we formulate a rougher version of the Composition Theorem that suffices for our purposes here.

§2.1. Finite fragments of theories. Let L be a first-order language and K be a class of structures for L. If $A \in K$ and $a_1, \ldots, a_k \in A$ let $\text{Th}^0(A, a_1, \ldots, a_k)$ be the quantifier-free type of the k-tuple $(a_1, \ldots, a_k)$ in the structure A. We would like to treat $\text{Th}^0(A, a_1, \ldots, a_k)$ as a finite constructive object.

Proviso. For every k the set

$$\{\text{Th}^0(A, a_1, \ldots, a_k) \colon A \in K \text{ and } a_1, \ldots, a_k \in A\}$$

is finite. Every $\text{Th}^0(A, a_1, \ldots, a_k)$ can be fully described in the theory of K by a single quantifier-free formula (in variables $v_1, \ldots, v_k$).

The last sentence of the Proviso means that for every $A, k, a_1, \ldots, a_k$ there is a quantifier-free formula $\phi(v_1, \ldots, v_k)$ such that $A \models \phi(a_1, \ldots, a_k)$ and if $B \in K$ and $b_1, \ldots, b_k \in B$ and $B \models \phi(b_1, \ldots, b_k)$ then $\mathrm{Th}^0(B, b_1, \ldots, b_k) = \mathrm{Th}^0(A, a_1, \ldots, a_k)$. We identify every $\mathrm{Th}^0(A, a_1, \ldots, a_k)$ with a quantifier-free formula that fully describes it. In our application the describing formulas will be constructed in a uniform way.

By induction on n we define *n-theories* $\mathrm{Th}^n(A, a_1, \ldots, a_k)$ where $A \in K$ and $a_1, \ldots, a_k \in A$. 0-theories have been defined already. Further,

$$\mathrm{Th}^{n+1}(A, a_1, \ldots, a_k) = \{\mathrm{Th}^n(A, a_1, \ldots, a_k, b): b \in A\}.$$

Let 0-k-Box be $\{\mathrm{Th}^0(A, a_1, \ldots, a_k): A \in K \text{ and } a_1, \ldots, a_k \in A\}$. Let $(n + 1)$-k-Box be the power set of n-$(k + 1)$-Box. It is easy to see that every $\mathrm{Th}^n(A, a_1, \ldots, a_k)$ belongs to n-k-Box.

Let us recall the definition of the *quantifier depth* $\mathrm{QD}(\phi)$ of a first-order formula ϕ. $\mathrm{QD}(\phi) = 0$ if ϕ is quantifier-free. If ϕ is a Boolean combination of formulas $\phi_1, \ldots, \phi_k$ then $\mathrm{QD}(\phi) = \max\{\mathrm{QD}(\phi_1), \ldots, \mathrm{QD}(\phi_k)\}$. If ϕ is $\exists v \psi(v)$ or $\forall v \psi(v)$ then $\mathrm{QD}(\phi) = 1 + \mathrm{QD}(\psi)$.

CLAIM 1. *For every n, k and every t in n-k-*Box *there is an L-formula $\phi_t(v_1, \ldots, v_k)$ of quantifier depth n such that for every $A \in K$ and every $a_1, \ldots, a_k \in A$, $A \models \phi_t(a_1, \ldots, a_k)$ iff* $\mathrm{Th}^n(A, a_1, \ldots, a_k) = t$.

PROOF (by induction on n). Our Proviso takes care of the case $n = 0$. If $t \in (n + 1)$-k-Box then ϕ_t is the conjunction of the formulas

$$\bigwedge_{s \in t} \exists v_{k+1}\, \phi_s(v_1, \ldots, v_{k+1}) \qquad \text{and} \qquad \forall v_{k+1} \bigvee_{s \in t} \phi_s(v_1, \ldots, v_{k+1}). \quad \square$$

CLAIM 2. *There is an algorithm that, given* $\mathrm{Th}^n(A, a_1, \ldots, a_k)$ *and an L-formula $\phi(v_1, \ldots, v_k)$ of quantifier depth n with free variables as shown, computes the truth value of $\phi(a_1, \ldots, a_k)$ in A.*

The proof is an easy induction on n. $\square$

Note that the quantifier depth of a prenex first-order formula is the length of its prefix. Here the prefix of a first-order formula is just a word in the alphabet $\{\forall, \exists\}$. Blocks of universal quantifiers and blocks of existential quantifiers alternate in a prefix. The *alternation type* of a prefix is the corresponding sequence of lengths of the blocks of quantifiers. For example, (6, 7, 4) is the alternation type of both $\forall^6\exists^7\forall^4$ and $\exists^6\forall^7\exists^4$. In order to prove the Composition Theorem for trees we need a generalization of the notion of n-theory that reflects not only the length of a prefix but also its alternation type.

In this paper ξ and η range over alternation types, i.e., over finite sequences of positive integers. By induction on the length $\mathrm{lh}(\xi)$ we define the *ξ-theory* of a structure $A \in K$ augmented by distinguished elements $a_1, \ldots, a_k$. The empty alternation type will be denoted simply 0. We put

$$0\text{-}\mathrm{Th}(A, a_1, \ldots, a_k) = \mathrm{Th}^0(A, a_1, \ldots, a_k).$$

If ξ is the extension $\eta^\frown m$ of η by an additional member m then

$$\xi\text{-}\mathrm{Th}(A, a_1, \ldots, a_k) = \{\eta\text{-}\mathrm{Th}(A, a_1, \ldots, a_k, b_1, \ldots, b_m): b_1, \ldots, b_m \in A\}.$$

We generalize the notion of n-k-Box as well. 0-k-Box is already defined.

If $\xi = \eta\hat{\ }m$ then ξ-k-Box is the power-set of η-$(k + m)$-Box. It is easy to see that every ξ-Th$(A, a_1, \ldots, a_k)$ belongs to ξ-k-Box. It will be convenient for us to order every ξ-k-Box in a standard manner. The order may be, for example, lexicographical.

Note that 3-theories are (1, 1, 1)-theories, not (3)-theories. To stress the distinction, we will use the notation 1^n for the sequence of n ones. Thus n-theories are 1^n-theories.

CLAIM 3. *There is an algorithm that computes an arbitrary* ξ-Th$(A, a_1, \ldots, a_k)$ *from* Th$^n(A, a_1, \ldots, a_k)$, *where n is the sum of members of* ξ.

The proof is easy.

§2.2. Compositions of trees. The alphabet $\{l, r\}$ will be called the *binary alphabet.* Words in the binary alphabet will be called *binary words.* The set $\{l, r\}^*$ of all binary words forms the *full binary tree* T. The empty word e is the *root* of T. Every binary word is a *node* of T. For every binary word x, the words xl and xr are the *left successor* and the *right successor* of x respectively, whereas x is the *predecessor* of both xl and xr. In this paper a *tree* is a nonempty subset of T closed under predecessors.

To put our discussion about trees into the framework of §2.1 we should specify the language L and the class K of structures in question. Let L be the first order language of Boolean algebras augmented by unary predicates Singleton, Root, End, LB (for Left Border), RB (for Right Border), and by binary predicates LS (for Left Successor), RS (for Right Successor). We suppose that L contains the usual Boolean operations and the equality predicate. L will be called in this paper the *monadic language of trees.* Every tree M gives a *standard model* for L in the following way. Consider the Boolean algebra of subsets of M and define additionally: Singleton(X) holds if X is singleton, Root(X) holds if $e \in X$, End(X) holds if X contains an end-point of M, LB(X) holds if no element of X has a left successor in M, and RB(X) holds if no element of X has a right successor in M; LS(X, Y) holds if there is $x \in M$ such that $X = \{x\}$ and $Y = \{xl\}$; and RS(X, Y) holds if there is $x \in M$ such that $X = \{x\}$ and $Y = \{xr\}$. Let K be the class of standard models for L. It is easy to see that our L, K satisfy the Proviso in §2.1. The quantifier-free type Th$^0(M, X_1, \ldots, X_k)$ of a k-tuple $(X_1, \ldots, X_k)$ of subsets of a tree M will be identified with the conjunction of formulas $\tau = 0$, Singleton(τ), Root(τ), End(τ), LB(τ), RB(τ), LS(σ, τ) and RS(σ, τ), where σ, τ are disjunctive normal terms in variables $v_1, \ldots, v_k$ and the formulas are true in M under the evaluation $v_1 = X_1, \ldots, v_k = X_k$.

REMARK. The reader may wonder why we composed this particular monadic language of trees. We did it in order to prove Lemma 1 in §2.3 and Lemma 1 in §3.1.

We define a composition of trees. Let M be a tree. A *grafting function* on M is a function g satisfying the following condition. Domain$(g) \subseteq M \times \{l, r\}$. If $(x, l) \in$ Domain(g) then x does not have a left successor in M, and if $(x, r) \in$ Domain(g) then x does not have a right successor in M. Every value $g(x, d)$ of g is a tree. (We use d to vary over $\{l, r\}$ because it is the first letter of the word "direction".) The *composition* of a tree M and a *grafting function* g on M is the tree

$$M \cup \{xdy\colon (x, d) \in \text{Domain}(g),\ y \in g(x, d)\}.$$

M is the *basis* and the *basic summand* of the composition. Every tree $g(x, d)$ is a *graft* and also a *summand* of the composition.

§2.3. The Composition Theorem. Let N be the composition of a tree M and a grafting function g on M. If $X \subseteq N$ let $X|M = X \cap M$ and $X|g(x, d) = \{y \in g(x, d): xdy \in X\}$ for every graft $g(x, d)$ of N. If $\bar{X}$ is a k-tuple $(X_1, \ldots, X_k)$ of subsets of N and S is a summand of N (a basic summand or a graft), let $\bar{X}|S = (X_1|S, \ldots, X_k|S)$.

For every alternation type ξ, every natural number k and every k-tuple $\bar{X}$ of subsets of N let $L^g(\xi, X)$ be the sequence

$$\langle L^g_t(\xi, \bar{X}): t \in \xi\text{-}k\text{-Box}\rangle$$

of subsets of M, where

$$L^g_t(\xi, \bar{X}) = \{x \in M: (x, l) \in \text{Domain}(g) \text{ and } \xi\text{-Theory}(g(x, l)), \bar{X}|g(x, l)) = t\}.$$

The sequence $R^g(\xi, \bar{X})$ of subsets of M is defined similarly.

LEMMA 1. *There is an algorithm that computes* $\mathrm{Th}^0(N, \bar{X})$ *from*

$$\mathrm{Th}^0(M, \bar{X}|M, L^g(0, \bar{X}), R^g(0, \bar{X})).$$

PROOF. Let $k = \mathrm{lh}(\bar{X})$ and $X_1, \ldots, X_k$ be the components of $\bar{X}$. Let $L = L^g(0, \bar{X}), R = R^g(0, \bar{X})$ and for every $t \in 0\text{-}k\text{-Box}$, let $L_t = L^g_t(0, \bar{X})$ and $R_t = R^g_t(0, \bar{X})$. If τ is a Boolean term in variables $v_1, \ldots, v_k$ let $\tau^* = \tau(X_1, \ldots, X_k)$ where the complements are computed in N. By induction on τ it is easy to check that for every summand S of N, $\tau^*|S = (X_1|S, \ldots, X_k|S)$ where the complements (on the right side) are taken in S.

In order to compute 0-Th$(N, \bar{X})$ it suffices to evaluate statements $\tau^* = 0$, Singleton(τ^*), Root(τ^*), End(τ^*), LB(τ^*), RB(τ^*), LS(σ^*, τ^*) and RS(σ^*, τ^*), where σ, τ are in the disjunctive normal form. For each of these statements, we provide a necessary and sufficient condition that is readily checkable when $\mathrm{Th}^0(M, \bar{X}|M, L, R)$ is given. Let s and t range over the set $\{t \in 0\text{-}k\text{-Box}: L_t \cup R_t \neq 0\}$.

$\tau^* = 0$ iff $\tau^*|M$ is empty and every t implies $\tau = 0$. τ^* is singleton iff either $\tau^*|M$ is singleton and every t implies $\tau = 0$ or else $\tau^*|M = 0$ and there is an s such that $L_s \cup R_s$ is singleton, s implies Singleton(τ) and every other t implies $\tau = 0$. $N \models \mathrm{Root}(\tau^*)$ iff $M \models \mathrm{Root}(\tau^*|M)$. $N \models \mathrm{End}(\tau^*)$ iff some t implies End(τ) or $M \models \mathrm{End}((\tau^*|M) - U)$ where U is the union of all sets L_t and R_t. $N \models \mathrm{LB}(\tau^*)$ if $M \models \mathrm{LB}(\tau^*|M)$, and $\tau^*|M$ avoids any L_t, and every t implies LB(τ). The case of RB is similar.

$N \models \mathrm{LS}(\sigma^*, \tau^*)$ iff both σ^* and τ^* are singleton and either $M \models \mathrm{LS}(\sigma^*|M, \tau^*|M)$ or some t implies LS(σ, τ) or else there is a t such that $0 \neq (\sigma^*|M) \subseteq L_t$ and t implies Root(τ^*). The case of RS is similar. □

By induction on the length of an alternation type ξ we define alternation types $H(\xi, k)$. If ξ is empty then every $H(\xi, k)$ is empty. If $\xi = \eta\hat{\ }j$ then

$$H(\xi, k) = H(\eta, k + j)\hat{\ }(j + 2\text{Cardinality } (\eta\text{-}(k + j)\text{-Box})).$$

THEOREM 2 (COMPOSITION THEOREM). *There is an algorithm* COMP *that for every*

ξ, k and every k-tuple $\bar{X}$ of subsets of N, computes ξ-Th$(N, \bar{X})$ *from* $H(\xi, k)$-Th$(M, \bar{X}|M, L^g(\xi, \bar{X}), R^g(\xi, \bar{X}))$.

PROOF. By induction on n we construct algorithms COMP_n such that every COMP_n does the job of COMP in the case length$(\xi) = n$. The construction is uniform in n and results in the desired algorithm COMP. Lemma 1 takes care of the case $n = 0$. Suppose that COMP_n is constructed already. Instead of defining COMP_{n+1} formally we just explain how it works.

Let $\xi = \eta\hat{\ }j$ be an alternation type of length $n + 1$. ξ-Th$(N, \bar{X})$ is the set of η-theories of structures $(N, \bar{X}\hat{\ }\bar{Y})$ where $\bar{Y}$ ranges over j-tuples of subsets of N. COMP_n will compute ξ-Th$(N, \bar{X})$ from the set S1 of $H(\eta, k + j)$-theories of structures $(M, \bar{X}|M, \bar{Y}|M, L^g(\eta, \bar{X}\hat{\ }\bar{Y}), R^g(\eta, \bar{X}\hat{\ }\bar{Y}))$ where $\bar{Y}$ ranges over j-tuples of subsets of N. S1 is computable from the set S2 of $H(\eta, k + j)$-theories of structures $(M, \bar{X}|M, L^g(\xi, \bar{X}), R^g(\xi, \bar{X}), \bar{Y}|M, L^g(\eta, \bar{X}\hat{\ }\bar{Y}), R^g(\eta, \bar{X}\hat{\ }\bar{Y}))$ where $\bar{Y}$ again ranges over j-tuples of subsets of N.

From the other side, the given $H(\xi, k)$-Th$(M, \bar{X}|M, L^g(\xi, \bar{X}), R^g(\xi, \bar{X}))$ is the set S3 of $H(\eta, k + j)$-theories of structures $(M, \bar{X}|M, L^g(\xi, \bar{X}), R^g(\xi, \bar{X}), \bar{Z}, \bar{U}, \bar{V})$ where $\bar{Z}$ ranges over j-tuples of subsets of M and $\bar{U}$, $\bar{V}$ range over tuples of subsets of M of length Cardinality (η-$(k + j)$-Box). Evidently S2 $\subseteq$ S3. Let

$$u = H(\eta, k + j)\text{-Th}(M, \bar{X}|M, L^g(\xi, \bar{X}), R^g(\xi, \bar{X}), \bar{Z}, \bar{U}, \bar{V})$$

be an element of S3. We give a checkable criterion for u to belong to S2: $u \in$ S2 iff (i) the sequence $\bar{U} = \langle U_t\colon t \in \eta\text{-}(k + j)\text{-Box}\rangle$ partitions $\bigcup\{L^g_s(\xi, \bar{X})\colon s \in \xi\text{-}k\text{-Box}\}$ and $t \in s$ whenever U_t meets $L^g_s(\xi, \bar{X})$, and (ii) the same for $\bar{V}$ and $R^g(\xi, \bar{X})$.

The "only if" direction is obvious. To prove the "if" direction suppose that u satisfies (i) and (ii). Choose a j-tuple $\bar{Y}$ of subsets of N such that $\bar{Y}|M = \bar{Z}$ and for every graft $S = g(x, l)$ (respectively $S = g(x, r)$), if $x \in U_t \cap L^g_s(\xi, \bar{X})$ (respectively $x \in V_t \cap R^g_s(\xi, \bar{X})$) then η-Th$(S, (\bar{X}|S)\hat{\ }(\bar{Y}|S)) = t$. Then u is the $H(\eta, j + k)$-theory of the structure

$$(M, \bar{X}|M, L^g(\xi, \bar{X}), R^g(\xi, \bar{X}), \bar{Y}|M, L^g(\eta, \bar{X}\hat{\ }\bar{Y}), R^g(\eta, \bar{X}\hat{\ }\bar{Y})),$$

hence it belongs to S2. $\square$

§2.4. Corollaries of the Composition Theorem. For every n and k, let $h(n, k)$ be the sum of members of the alternation type $H(1^n, k)$. Here 1^n is the sequence of n ones, and H is defined in §2.3.

THEOREM 1 (SECOND COMPOSITION THEOREM). *There is an algorithm* COMP2 *such that for every M, g, N, n, k, $\bar{X}$, if N is the composition of M, g and if $\bar{X}$ is a k-tuple of subsets of N then* COMP2 *computes* $\mathrm{Th}^n(N, \bar{X})$ *from*

$$\mathrm{Th}^{h(n,k)}(M, \bar{X}|M, L^g(1^n, \bar{X}), R^g(1^n, \bar{X})).$$

PROOF. Combine the algorithm COMP of the Composition Theorem and the algorithm of Claim 3 in §2.1. $\square$

Let $h^*(n, k) = h(n, k) + 2$ Cardinality (1^n-k-Box).

THEOREM 2. *Suppose that A, B are trees, $\bar{X}$ is a k-tuple of subsets of A, $\bar{Y}$ is a k-tuple of subsets of B, and the structures $(A, \bar{X})$, $(B, \bar{X})$ are $h^*(n, k)$-equivalent. Suppose that f is a grafting function on A, M is the composition of A and f, and S is a* *set of trees such that for every $C \in$* Range(f) *there is a $C' \in S$ that is n-equivalent to C. Then there is a grafting function g on B such that* Range(g) $\subseteq S$ *and if N is the composition of B and g then the structures $(M, \tilde{X})$ and $(N, \tilde{Y})$ are n-equivalent.*

PROOF. There are sequences

$$L = \langle L_t \colon t \in 1^n\text{-}k\text{-Box}\rangle, \qquad R = \langle R_t \colon t \in 1^n\text{-}k\text{-Box}\rangle$$

such that the structure $(B, \tilde{Y}, L, R)$ is $h(n, k)$-equivalent to $(A, \tilde{X}, L^f(1^n, \tilde{X}), R^f(1^n, \tilde{Y}))$. In particular $B \models \mathrm{LB}(L_t)$ and $B \models \mathrm{RB}(R_t)$ for every t. Note that $L^f(1^n, \tilde{X}) = L^f(1^n, \tilde{0})$ and $R^f(1^n, \tilde{X}) = R^f(1^n, \tilde{0})$ where $\tilde{0}$ is the k-tuple of empty sets. This happens because $\tilde{X}|f(x, d)$ is empty for every $(x, d) \in \mathrm{Domain}(f)$.

We build the desired grafting function g in such a way that $L^g(L^n, \tilde{0}) = L$ and $R^g(1^n, \tilde{0}) = R$. Given $x \in L_t$ pick $y \in L^f_t(1^n, \tilde{0})$ and $C' \in S$ such that C' is n-equivalent to $f(x, l)$; then set $g(x, l) = C'$. Define the values $g(x, r)$ similarly. It is easy to see that $L = L^g(1^n, \tilde{Y})$ and $R = R^g(1^n, \tilde{Y})$. By the Second Composition Theorem, $\mathrm{Th}^n(M, \tilde{X}) = \mathrm{Th}^n(N, \tilde{Y})$. $\square$

CHAPTER 3. INTERVALS

In §3.1 we prove another composition theorem; this time it is about addition of intervals of the full binary tree. In §3.2 we introduce and study n-extensible intervals. These intervals are too diverse and lengthy to be described by their n-theories. They will play a prominent role in the sequel.

§3.1. Addition. We order binary words as follows: $x \leq y$ if x is an initial segment of y. Recall that in this paper a tree is a subset of the full binary tree $T = \{l, r\}^*$. A tree A will be called an *interval* if there is a binary word a with $\{x \colon x \leq a\} = A$. If $x \leq y$ let $[x, y] = \{z \colon x\, z \leq y\}$. In other words, $[x, y]$ is the interval $[e, a]$ where $xa = y$. Next we define addition of intervals: $[e, a] + [e, b] = [e, ab]$. In particular, $[x, y] + [y, z] = [x, z]$ for any x, y, z. The addition is associative.

In the rest of this section A is an interval $[e, a]$ and B is an interval $[e, b]$. Let C be the interval $[e, ab]$. For every $X \subseteq C$ let $X|A = X \cap A$ and $X|B = \{x \colon ax \in X\}$. If $\tilde{X}$ is a k-tuple $(X_1, \ldots, X_k)$ and S is either A or B let $\tilde{X}|S = (X_1|S, \ldots, X_k|S)$.

LEMMA 1. *There is an algorithm that computes* $\mathrm{Th}^0(C, \tilde{X})$ *from* $\mathrm{Th}^0(A, \tilde{X}|A)$ *and* $\mathrm{Th}^0(B, \tilde{X}|B)$.

PROOF. Let $k = \mathrm{lh}(\tilde{X})$; let $X_1, \ldots, X_k$ be the components of $\tilde{X}$. If $\tau(v_1, \ldots, v_k)$ is a Boolean term in variables $v_1, \ldots, v_k$ let τ^* be the value $\tau(X_1, \ldots X_k)$ computed in C. By induction on τ it is easy to check that if $S = A$ or $S = B$ then $\tau^*|S$ is the value $\tau(X_1|S, \ldots, X_k|S)$ computed in S. In order to compute $\mathrm{Th}^0(C, \tilde{X})$ it suffices to evaluate the statements $\tau^* = 0$, Singleton(τ^*), Root(τ^*), End(τ^*), LB(τ^*), RB(τ^*), LS(σ^*, τ^*) and RS(σ^*, τ^*), where σ, τ are in the disjunctive normal form, when $\mathrm{Th}^0(A, \tilde{X}|A)$ and $\mathrm{Th}^0(B, \tilde{X}|B)$ are given.

$\tau^* = 0$ iff both $\tau^*|A$ and $\tau^*|B$ are empty. τ^* is singleton iff either $\tau^*|A$ is singleton and $\tau^*|B$ is empty, or $\tau^*|A$ is empty and $\tau^*|b$ is singleton, or else both $\tau^*|A$ and $\tau^*|B$ are singleton and $A \models \mathrm{End}(\tau^*|A)$ and $B \models \mathrm{Root}(\tau^*|B)$. Further, $C \models \mathrm{Root}(\tau^*)$ iff $A \models \mathrm{Root}(\tau^*|A)$; $C \models \mathrm{End}(\tau^*)$ iff $B \models \mathrm{End}(\tau^*|B)$; $C \models \mathrm{LB}(\tau^*)$ iff $A \models \mathrm{LB}(\tau^*|A)$ and $B \models \mathrm{LB}(\tau^*|B)$ (the case of RB is similar); and $C \models \mathrm{LS}(\sigma^*, \tau^*)$ iff both σ^*, τ^* are singleton and either $A \models \mathrm{LS}(\sigma^*|A, \tau^*|B)$ or $B \models \mathrm{RS}(\sigma^*|A, \tau^*|B)$ (the case of RS is similar). $\square$

THEOREM 2 (ADDITION THEOREM). *There is an algorithm* PLUS *that computes* $\mathrm{Th}^n(C, \bar{X})$ *from* $\mathrm{Th}^n(A, \bar{X}|A)$ *and* $\mathrm{Th}^n(B, \bar{X}|B)$ *for every n, k and every k-tuple $\bar{X}$ of subsets of C.*

PROOF. By induction on n we construct algorithms PLUS_n such that PLUS_n computes $\mathrm{Th}^n(C, \bar{X})$ from $\mathrm{Th}^n(A, \bar{X}|A)$ and $\mathrm{Th}^n(B, \bar{X}|B)$ for every k and $\bar{X}$. The construction is uniform in n and results in the desired algorithm PLUS. Lemma 1 takes care of the case $n = 0$. Suppose that PLUS_n is constructed already. To simplify notation we suppose that $k = 0$. Then

$$\begin{aligned}\mathrm{Th}^{n+1}(C) &= \{\mathrm{Th}^n(C, X): X \subseteq C\} = \{\mathrm{PLUS}_n(\mathrm{Th}^n(A, X|A), \mathrm{Th}^n(B, X|B)): X \subseteq C\}\\ &= \{\mathrm{PLUS}_n(s, t): s \in \mathrm{Th}^{n+1}(A),\ t \in \mathrm{Th}^{n+1}(B) \text{ and if } s = \mathrm{Th}^n(A, Y),\\ &\qquad\qquad t = \mathrm{Th}^n(B, Z), \text{ then } A \models \mathrm{End}(Y) \text{ iff } B \models \mathrm{Root}(Z)\}\\ &= \mathrm{PLUS}_{n+1}(\mathrm{Th}^{n+1}(A), \mathrm{Th}^{n+1}(B)). \quad \square\end{aligned}$$

If $s = \mathrm{PLUS}(t_1, t_2)$ we will write $s = t_1 + t_2$.

COROLLARY 3. *For all binary words $x \le y \le z$ and for every n,*

$$\mathrm{Th}^n[x, z] = \mathrm{Th}^n[x, y] + \mathrm{Th}^n[y, z].$$

§3.2. Extensible intervals. An interval $[x, y]$ will be called *n-extensible* if for every $y_1, y_2 \ge y$ there is a $z \ge y_1$ with $\mathrm{Th}^n[x, z] = \mathrm{Th}^n[x, y_2]$.

CLAIM 1. *An interval $[x, y]$ is n-extensible iff for every $y_1 \ge y$,*

$$\{\mathrm{Th}^n[x, z]: z \ge y_1\} = \{\mathrm{Th}^n[x, y_2]: y_2 \ge y\}.$$

The proof is clear.

CLAIM 2. *Suppose $u \le x$. For every n there is a binary word $y \ge x$ such that $[u, y]$ is n-extensible.*

PROOF. Let $F(y) = \{\mathrm{Th}^n[u, z]: z \ge y\}$ for $y \ge x$. Values of F are finite sets. F is monotone and nonincreasing, i.e. $F(y') \subseteq F(y)$ if $y \le y'$. Hence there is a y such that $F(y) = F(y')$ for every $y' \ge y$. By Claim 1 $[u, y]$ is n-extensible. $\square$

CLAIM 3. *Suppose that $u \le w \le x \le y$ and $[w, x]$ is n-equivalent to an n-extensible interval $[w', x']$. Then $[u, y]$ is n-extensible.*

PROOF. Given $y_1, y_2 \ge y$ we seek $z \ge y_1$ such that $[u, z]$ is n-equivalent to $[u, y_2]$. There are a, b with $y_1 = xa$, $y_2 = xb$. Let $y_1' = x'a$. $y_2' = x'b$. Since $[w', x']$ is n-extensible there are z' and c such that $z' = y_1'c$ and $[w', z']$ is n-equivalent to $[w', y_2']$. Set $z = y_1c$. We use the sign $\sim$ to indicate n-equivalence. Then

$$\begin{aligned}[w, z] &= [w, x] + [x, z] = [w, x] + [e, ac]\\ &\sim [w', x'] + [e, ac] = [w', x'] + [x', z'] = [w', z']\\ &\sim [w', y_2'] = [w', x'] + [x', y_2'] = [w', x'] + [e, b]\\ &\sim [w, x] + [e, b] = [w, x] + [x, y_2] = [w, y_2].\end{aligned}$$

Hence $[u, z] = [u, w] + [w, z] \sim [u, w] + [w, y_2] = [u, y_2]$. $\square$

CHAPTER 4. MERGING

In §4.1 we introduce n-samples. They are finite sets of binary words that are so far spread that their n-theories cannot describe them properly. The union of two

n-samples may not be an n-sample. There is, however, a less direct way to merge two n-samples. The Merging Theorem, proved in §4.2, takes care of that. It is the main theorem of this section.

§4.1. **Elongation.** For any binary words x, y let $x \wedge y$ be the longest common prefix for x, y. We will say that x is *n-older* than y if $[x \wedge y, x]$ is neither singleton nor n-extensible whereas $[x \wedge y, y]$ is n-extensible. The reasson why we close the word "older" will be cleaar when we come to forcing.

CLAIM 1. *For every n, the relation n-older is transitive.*

PROOF. Suppose that x is n-older than y and y is n-older than z (see Figure 1).

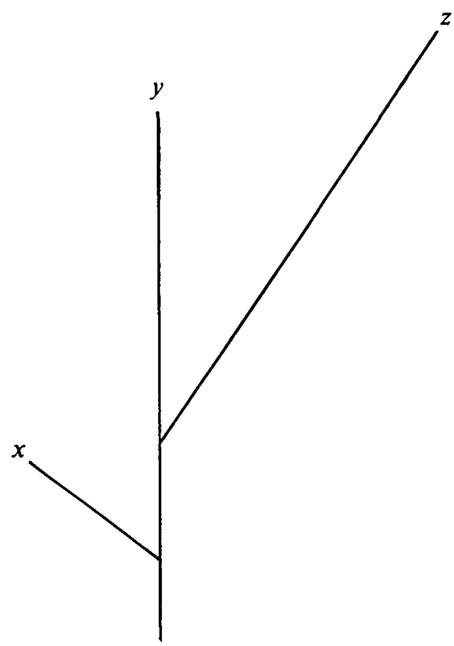

FIGURE 1

Then $[x \wedge y, y]$ is n-extensible whereas $[y \wedge z, y]$ is not. By Claim 3 in §3.2, $x \wedge y < y \wedge z$. Hence $x \wedge y = x \wedge z$. Thus $[x \wedge z, x]$ is equal to $[x \wedge y, x]$, which is neither singleton nor n-extensible. The interval $[x \wedge z, z]$ includes an n-extensible interval $[y \wedge z, z]$. By Claim 3 in §3.2, $[y \wedge z, z]$ is n-extensible. $\square$

A finite set X of binary words will be called an *n-sample* if

(i) for every $x \in X$, the interval $[e, x]$ is n-extensible, and

(ii) for every $x, y \in X$, either x is n-older than y or y is n-older than x or else both intervals $[x \wedge y, x]$ and $[x \wedge y, y]$ are n-extensible.

If X is a finite set of binary words, let Bush(X) (the *bush* of X) be the smallest tree that includes X. Recall that the function h^* was defined in §2.4.

THEOREM 2 (ELONGATION THEOREM). *Suppose that* $j = h^*(i, 0)$ *and* X *is a j-sample. For every* $m \geq 1$ *there is a j-sample* Y *such that*:

1. Bush(Y) *is i-equivalent to* Bush(X),
2. *if* $u < x \in Y$ *and* $\mathrm{lh}(u) \leq m$ *then* $[u, x]$ *is j-extensible, and*

3. *if x, y are distinct elements of Y and $u \leq x \wedge y$ then there is $d \in \{l, r\}$ such that $[ud, z]$ is j-extensible for every $ud \leq z \in Y$.*

PROOF (by induction on the cardinality $|X|$ of X). The case $|X| = 0$ is trivial.

Case $|X| = 1$. Let a be the only element of X. Choose $x \geq a$ of length at least m. By Claim 2 in §3.2 there is $y \geq x$ such that $[x, y]$ is j-extensible. Since $[e, a]$ is j-extensible there is $b \geq y$ such that $[e, b]$ is j-equivalent to $[e, a]$. $Y = \{b\}$ is the desired j-sample.

Case $|X| > 1$. Let a be a j-oldest element of X. Bush(X) is the composition of $[e, a]$ and a certain grafting function f on $[e, a]$. We build the bush of the desired j-sample Y as the composition of some extension $[e, b]$ of $[e, a]$ and some grafting function g on $[e, b]$.

Extending $[e, a]$. As in the case $|X| = 1$ above, build $b \geq a$ such that $[e, b]$ is j-equivalent to $[e, a]$ and for every $u \leq b$, if $\mathrm{lh}(u) \leq m$ then $[u, b]$ is j-extensible.

Building g. It is easy to see that every tree in Range(f) is the bush of some j-sample. By the induction hypothesis, for every Bush(X_1) in Range(f) there is a j-sample Y_1 such that Bush(Y_1) is i-equivalent to Bush(X_1) and Y_1 satisfies the properties 2 and 3 stated in the Elongation Theorem. Let

$$S = \{\mathrm{Bush}(Y_1) \colon \mathrm{Bush}(X_1) \in \mathrm{Range}(f)\}.$$

By Theorem 2 in §2.4 there is a grafting function g on $[e, b]$ such that Range(g) $\subseteq S$ and the composition N of $[e, b]$ and g is i-equivalent to Bush(X).

There is a Y such that $N = \mathrm{Bush}(Y)$. This is the desired j-sample. □

§4.2. **Merging.**

THEOREM 1. *Suppose that $\{a\}$ is a j-sample and $m = \mathrm{lh}(a)$. Suppose that Y is a j-sample and Y satisfies the properties 2 and 3 stated in the Elongation Theorem. Then there is a $b \geq a$ such that $[e, b]$ is j-equivalent to $[e, a]$, and $\{b\} \cup Y$ is a j-sample, and for every $y \in Y$, both $[b \wedge y, b]$ and $[b \wedge y, y]$ are j-extensible.*

PROOF. Let $X = \{x \wedge y \colon x, y$ are distinct elements of $Y\}$. Build a sequence $\langle a_i \colon 0 \leq i \leq k \rangle$ as follows. Set $a_0 = a$. Suppose a_i is built already. If $a_i \notin \mathrm{Bush}(X)$ set $k = i$ and stop. If $a_i \in \mathrm{Bush}(X)$ then, by property 3, there is a $d_i \in \{l, r\}$ such that $[a_i d_i, z]$ is j-extensible for every $a_i d_i \leq z \in Y$. Set $a_{i+1} = a_i d_i$.

Set $a' = a_k d'$ where $d' \in \{l, r\}$ and $a' \notin \mathrm{Bush}(Y)$. By Lemma 2 in §3.2 there is $a'' > a'$ such that $[a', a'']$ is j-extensible. Since $[e, a]$ is j-extensible, there is $b \geq a''$ such that $[e, b]$ is j-equivalent to $[e, a]$.

In order to prove that b is the desired extension of a it suffices to check that for every $y \in Y$, both $[b \wedge y, b]$ and $[b \wedge y, y]$ are j-extensible. $[b \wedge y, b]$ extends a j-extensible interval $[a', a'']$. By Claim 3 in §3.2, $[b \wedge y, b]$ is extensible. If $\mathrm{lh}(b \wedge y) \leq m$ then $[b \wedge y, y]$ is j-extensible by virtue of property 2. Otherwise $b \wedge y = a_{i+1}$ for some $i \leq k$, and $[b \wedge y, y]$ is j-extensible by the choice of a_{i+1}. □

COROLLARY 2. *Suppose that $j = h^*(i, 0)$ and both $\{a\}$ and X are j-samples. Then there are $b \geq a$ and a j-sample Y such that $[e, b]$ is j-equivalent to $[e, a]$,* Bush(Y) *is i-equivalent to* Bush(X), *$\{b\} \cup Y$ is a j-sample, and for every $y \in Y$, both $[b \wedge y, b]$ and $[b \wedge y, y]$ are j-extensible.*

PROOF. Just combine the Elongation Theorem (with $m = \mathrm{lh}(a)$) and Theorem 1. □

THEOREM 3 (MERGING THEOREM). *Suppose that* $j = h^*(i, 0)$ *and* X, Y *are* j*-samples. There are* j*-samples* X', Y' *such that* Bush(X') *is* i*-equivalent to* Bush(X), Bush(Y') *is* i*-equivalent to* Bush(Y), $X' \cup Y'$ *is a* j*-sample, and for every* $x \in X'$, $y \in Y'$, *both* $[x \wedge y, x]$ *and* $[x \wedge y, y]$ *are* j*-extensible.*

PROOF (by induction on $|X|$). The case $|X| = 0$ is trivial. Corollary 2 takes care of the case $|X| = 1$. Suppose $|X| > 1$.

Let a be a j-oldest element in X. By Corollary 2 there are $b \geq a$ and a j-sample Y_1 such that $[e, b]$ is j-equivalent to $[e, a]$, Bush(Y_1) is i-equivalent to Bush(Y), $\{b\} \cup Y_1$ is a j-sample, and for every $y \in Y_1$, both $[b \wedge y, b]$ and $[b \wedge y, y]$ are j-extensible.

Bush(X) is the composition of $[e, a]$ and a certain grafting function f_a. By Theorem 2 in §2.4 there is a grafting function f_b on $[e, b]$ such that Range$(f_b) \subseteq$ Range(f_a) and the composition of $[e, b]$ and f_b is i-equivalent to Bush(X). The composition of $[e, b]$ and f_b is Bush(X_1) for some j-sample X_1.

Let $[e, c] = [e, b] \cap$ Bush(Y_1). Bush(X_1) and Bush(Y_1) are the compositions of $[e, c]$ and certain grafting functions f_1, g_1 respectively. Without loss of generality $cl \leq b$ and $cr \in$ Bush(Y_1). We built new grafting functions f, g on $[e, c]$ in such a way that the compositions of $[e, c]$ with f, g are the bushes of the desired j-samples X', Y'.

Let Domain(f), Domain(g) be equal to Domain(f_1), Domain(g_1) respectively. If (x, d) belongs to Domain(f_1) $-$ Domain(g_1) set $f(x, d) = f_1(x, d)$. If (x, d) belongs to Domain(g_1) $-$ Domain(f_1) set $g(x, d) = g_1(x, d)$. Suppose that (x, d) belongs to both Domain(f_1) and Domain(g_1). Note that $d = r$ if $x = c$. It is easy to see that $f_1(x, d)$ and $g_1(x, d)$ are the bushes of some j-samples U, V respectively. It is easy to see that Bush$(U) \in$ Range$(f_b) \subseteq$ Range(f_a). Hence $|U| < |X|$. By the induction hypothesis there are j-samples U', V' such that Bush(U') is i-equivalent to Bush(U), Bush(V') is i-equivalent to Bush(V), $U' \cup V'$ is a j-sample, and for every $x \in U'$ $y \in V'$, both $[x \wedge y, x]$ and $[x \wedge y, y]$ are j-extensible. Set $f(x, d) = U'$ and $g(x, d) = V'$.

It is easy to see that the composition of $[e, c]$ and f (respectively g) is the bush of some j-sample X'(respectively Y'). By the Second Composition Theorem Bush(X') is i-equivalent to Bush(X_1), and Bush(Y') is i-equivalent to Bush(Y_1). It is easy to check that X', Y' are the desired j-samples. □

CHAPTER 5. FORCING

In this chapter we use forcing to prove the main theorem of this paper. The main theorem is stated in §5.1 and is proved in §5.3. We suppose that the reader has some knowledge about forcing. The paper Shoenfield [1971] suffices for our purposes.

§5.1. The Main Theorem. Let $\phi(v_1, v_2)$ be a formula in the monadic language of trees saying that v_1 is not empty and v_2 is a singleton subset of v_1. Evidently the full binary tree T satisfies the sentence $\forall v_1 \exists v_2 \phi(v_1, v_2)$.

As usual, ZFC is Zermelo-Fraenkel set theory with the axiom of choice. Models of ZFC will be called *worlds*.

THEOREM 1 (MAIN THEOREM). *Let* $\phi^*(v_1, v_2)$ *be a formula in the monadic language of trees with free variables as shown. Let* W *be a world. Suppose that, in* W,

(A) $T \models \forall v_1 \forall v_2(\phi^*(v_1, v_2) \rightarrow \phi(v_1, v_2))$, *and*

(B) $T \models$ (*for every* v_1 *there is a unique* v_2 *with* $\phi^*(v_1, v_2)$).

Then there is another world W' *where* ϕ^* *fails to satisfy* (A) *or* (B).

Theorem 1 will be proved in §5.3.

COROLLARY 2. *There is no formula* $\phi^*(v_1, v_2)$ *in the monadic language of trees that satisfies the conditions* (A) *and* (B) *in any world.*

PROOF OF COROLLARY 2. Rabin [1969] constructed a decision procedure for the monadic second-order theory of the full binary tree T. (See a simpler decision procedure for the same theory in Gurevich and Harrington [1982].) The decision procedure works in ZFC; it does not use any extra set-theoretic assumptions. If ϕ^* satisfies (A) and (B) in some world W, the decision procedure verifies (A) and (B) in W. It verifies therefore that (A) and (B) follow from ZFC, which contradicts Theorem 1. $\square$

§5.2. The forcing notion. Let $\alpha < \beta < \gamma < \delta$ be natural numbers. The number α will be specified in the next section. About β, γ and δ we suppose only that they are large enough to meet all requirements of this and the next sections. There will be only a handful of requirements and each will have the form

$\beta \geq$ (a number depending on α), or

$\gamma \geq$ (a number depending on α, β), or

$\delta \geq$ (a number depending on α, β, γ).

Let W be our ground world. Our forcing notion P consists of all δ-samples. In other words, δ-samples are our forcing conditions. We say that a forcing condition q is stronger than a forcing condition p (symbolically $p \leq q$) if $p \subseteq q$ and no $y \in q$ is δ-older than any $x \in p$. In the sequel p and q (with or without subscripts) range over P. We do not distinguish between elements of W and their canonical forcing names. Let $U = \{(x, p): p \in P \text{ and } x \in p\}$. Evidently U is a name for the union of the generic filter.

THEOREM 1. *For* $i = 1, 2$ *suppose that* p_i *is a forcing condition,* $A_i \subseteq p_i$, *and* p_i *forces* $\mathrm{Th}^\alpha(T, U, A_i) = t_i$. *Suppose that the structures* $(\mathrm{Bush}(p_i), A_i)$ *are* β-*equivalent. Then* $t_1 = t_2$.

PROOF. If G_i is a generic filter over P that contains p_i then $\mathrm{Th}^\alpha(T, \bigcup G_i, A_i) = t_i$ in $W(G_i)$. We build generic filters G_1, G_2 such that $p_1 \in G_1$, $p_2 \in G_2$ and $W(G_1) = W(G_2)$. Then we prove that the structures $(T, \bigcup G_1, A_1)$ and $(T, \bigcup G_2, A_2)$ are α-equivalent.

For every $i = 1, 2$ let g_i be the grafting function on $\mathrm{Bush}(p_i)$ that grafts a copy of T into each "bud" of $\mathrm{Bush}(p_i)$. Thus the composition of $\mathrm{Bush}(p_i)$ and g_i is equal to T. For every forcing condition q let $p_i * q$ be

$$\{xdy: (x, d) \in \mathrm{Domain}(g_i),\ x \notin p_i,\ y \in q\}.$$

It is easy to see that $p_i * q$ is a forcing condition. $\mathrm{Bush}(p_i * q)$ is the result of grafting a copy of $\mathrm{Bush}(q)$ into each "bud" $(x, d) \in \mathrm{Domain}(g_i)$ with $x \notin p_i$.

Let G be an arbitrary generic filter over P, and $G_i = \{p \in P: p \leq p_i * q \text{ for some } q \in G\}$. Clearly G_1, G_2 are generic filters over P and $W(G_1) = W(G_2) = W(G)$. Let us work now in $W(G)$.

The algorithm COMP2 of §2.4 computes $\mathrm{Th}^\alpha(T, \bigcup G_i, A_i)$ from the $h(\alpha, 2)$-theory of the structure

$$M_i' = (\mathrm{Bush}(p_i), p_i, A_i, L^i, R^i),$$

where $L^i = L^{g_i}(1^\alpha, \bigcup G_i, A_i)$ and $R^i = R^{g_i}(1^\alpha, \bigcup G_i, A_i)$. Thus it suffices to prove that M_1' and M_2' are $h(\alpha, 2)$-equivalent.

Let $M_i = (\mathrm{Bush}(p_i), A_i)$. It is easy to see that the same formula defines p_1 in M_1 and defines p_2 in M_2. The formula says that p_i is the set of end-points. Recall that $L^i = \langle L_t^i : t \in 1^\alpha\text{-2-Box}\rangle$ where L_t^i abbreviates $L_t^{g_i}(1^\alpha, \bigcup G_i, A_i)$, and similarly for R^i. It is easy to see that for every t the same formula defines L_t^1 in M_1 and defines L_t^1 in M_2, and similarly for R^i. If $t = \mathrm{Th}^\alpha(T, 0, 0)$ then $L_t^i = p_i$ and we can use the previous formula. If $t = \mathrm{Th}^\alpha(T, \bigcup G, 0)$ then the formula says that L_t^i is the set of elements $x \in \mathrm{Bush}(p_i) - p_i$ such that x does not have a left successor in $\mathrm{Bush}(p_i)$. In other cases the formula says $L_t^i = 0$. Since M_1 and M_2 are β-equivalent and β is sufficiently larger than $h(\alpha, 2)$ we have that M_1' and M_2' are $h(\alpha, 2)$-equivalent. □

THEOREM 2. *Suppose that p_1, p_2 are forcing conditions, $A_1 \subseteq p_1$, $A_2 \subseteq p_2$, and $(\mathrm{Bush}(p_1), A_1)$, $(\mathrm{Bush}(p_2), A_2)$ are γ-equivalent. Suppose also that p_1 forces $\mathrm{Th}^\alpha(T, U, A_1) = t_1$ for some t_1. Then p_2 forces $\mathrm{Th}^\alpha(T, U, A_2) = t_1$.*

PROOF. Take $q_2 \geq p_2$ such that q_2 forces $\mathrm{Th}^\alpha(T, U, A_2) = t_2$ for some t_2. It suffices to prove that $t_2 = t_1$.

Let g_2 be the grafting function on $\mathrm{Bush}(p_2)$ such that the composition of $\mathrm{Bush}(p_2)$ and g_2 is equal to $\mathrm{Bush}(q_2)$. Require $\gamma \geq h^*(\beta, 1)$. By Theorem 2 in §2.4 (with $n = \beta$ and $k = 1$) there is a grafting function g_1 on $\mathrm{Bush}(p_1)$ such that $\mathrm{Range}(g_1) \subseteq \mathrm{Range}(g_2)$ and if C is the composition of $\mathrm{Bush}(p_1)$ and g_1 then (C, A_1) is β-equivalent to $(\mathrm{Bush}(q_2), A_2)$. It is easy to see that $C = \mathrm{Bush}(q_1)$ for some forcing condition $q_1 \geq p_1$. Clearly, q_1 forces $\mathrm{Th}^\alpha(T, U, A_1) = t_1$. By Theorem 1, $t_1 = t_2$. □

THEOREM 3. *The empty condition forces $\mathrm{Th}^\alpha(T, U) = s$ for some s.*

PROOF. Suppose that forcing conditions p_1, p_2 force $\mathrm{Th}^\alpha(T, U) = t_1$, $\mathrm{Th}^\alpha(T, U) = t_2$ respectively. It suffices to prove that $t_1 = t_2$.

By the Merging Theorem (with $i = \gamma$ and $j = \delta$) there are forcing conditions q_1, q_2 such that $\mathrm{Bush}(q_1)$, $\mathrm{Bush}(q_2)$ are β-equivalent to $\mathrm{Bush}(p_1)$, $\mathrm{Bush}(p_2)$ respectively and $q_1 \cup q_2$ is a forcing condition stronger then each of q_1 and q_2. By Theorem 2, q_1, q_2 force $\mathrm{Th}^\alpha(T, U) = t_1$, $\mathrm{Th}^\alpha(T, U) = t_2$ respectively. But $q_1 \cup q_2$ forces both $\mathrm{Th}^\alpha(T, U) = t_1$ and $\mathrm{Th}^\alpha(T, U) = t_2$. Hence, $t_1 = t_2$. □

§5.3. Proving the Main Theorem. In this section we prove Theorem 1 of §5.1. Suppose that a formula $\phi^*(v_1, v_2)$ satisfies the conditions (A), (B) in some world W. We specify the number α of §5.2 as the quantifier depth of ϕ^*. We intend to prove that ϕ^* fails to satisfy (A) or (B) in any world $W(G)$ where G is a generic filter over P. Suppose the contrary. Then there is p forcing $T \models \phi^*(U, \{a\})$ for some binary word a. We suppose that p is a minimal forcing condition that forces $T \models \phi^*(U, \{a\})$. In particular p forces $a \in U$.

CLAIM 1. *The condition p contains a.*

PROOF. If p contains some $b < a$ then there is no $q \geq p$ that contains a. Hence

p forces $a \notin U$ (recall that $U = \{(x, q): x \in q \text{ and } q \in P\}$) which is impossible. If p does not contain any $b \leq a$ then some $p' \geq p$ contains a word $b > a$. Hence there is no $q \geq p'$ that contains a. Hence p' forces $a \notin U$ which is impossible. The only remaining possibility is that p contains a. □

CLAIM 2. *Every element of $p - \{a\}$ is δ-older than a.*

PROOF. Let $p_0 = \{x \in p: \text{either } x = a \text{ or } x \text{ is } \delta\text{-older than } a\}$. By virtue of the minimality of p it suffices to prove that p_0 forces $T \models \phi^*(U, \{a\})$. Let G be a generic filter over P that contains p_0. It suffices to prove that $T \models \phi^*(\bigcup G, \{a\})$ in $W(G)$. We work in $W(G)$.

Let g be the grafting function on Bush(p_0) such that the composition of Bush(p_0) and g is equal to T. The function g grafts a copy of T into each "bud" of Bush(p_0). The algorithm COMP2 of §2.4 computes $\mathrm{Th}^\alpha(T, \bigcup G, \{a\})$ from the $h(\alpha, 2)$-theory of the structure $M = (\mathrm{Bush}(p_0), p_0, \{a\}, L, R)$ where $L = L^g(1^\alpha, \bigcup G, \{a\})$ and $R = R^g(1^\alpha, \bigcup G, \{a\})$. Note that $L = L^g(1^\alpha, \bigcup G, 0)$ and $R = R^g(1^\alpha, \bigcup G, 0)$ because $\{a\}|g(x, d)$ is empty for every $(x, d) \in \mathrm{Domain}(g)$.

It is easy to see that every $(\bigcup G)|g(x, d)$ is a generic filter over P. By Theorem 3 in §5.2, $\mathrm{Th}^\alpha(g(x, d), (\bigcup G)|g(x, d))$ does not depend on G. Hence $\mathrm{Th}^\alpha(g(x, d), (\bigcup G)| g(x, d), 0)$ does not depend on G. Hence the sequences L, R do not depend on G. Hence $\mathrm{Th}^\alpha(T, \bigcup G, \{a\})$ does not depend on G. However $\mathrm{Th}^\alpha(T, \bigcup G, \{a\})$ implies $\phi^*(\bigcup G, \{a\})$ if G contains p. Hence $\mathrm{Th}^\alpha(T, \bigcup G, \{a\})$ implies $\phi^*(\bigcup G, \{a\})$ for every G that contains p_0. □

If $p = \{a\}$ let $B = \{e\}$. If $p - \{a\}$ is not empty let $B = \mathrm{Bush}(p - \{a\})$. Let b be the longest prefix of a that belongs to B. By Claim 2 in §3.2 there are $a' > al$ and $a'' > ar$ such that $[a, a']$ and $[a, a'']$ are δ-extensible. By Claim 2, $[b, a]$ is δ-extensible. Hence there are $a_1 \geq a'$ and $a_2 \geq a''$ such that $[b, a_1]$ and $[b, a_2]$ are δ-equivalent to $[b, a]$. Let p_1, p_2 be obtained from p by replacing a by a_1, a_2 respectively. It is easy to see that p_1, p_2 and $p_1 \cup p_2$ are forcing conditions.

CLAIM 3. *Both* $\mathrm{Bush}(p_1, \{a_1\})$ *and* $\mathrm{Bush}(p_2, \{a_2\})$ *are γ-equivalent to* $\mathrm{Bush}(p, \{a\})$.

PROOF. To uniformize notation let $a_0 = a$ and $p_0 = p$. Let $C_i = \mathrm{Bush}(p_i)$ for $i = 0, 1, 2$. Every C_i is the composition of B and some grafting function g_i on B. The algorithm COMP2 computes $\mathrm{Th}^\gamma(C_i, \{a_i\})$ from the $h(\gamma, 1)$-theory of the structure $M_i = (B, \{b_i\}, L^i(1^\gamma, \{b_i\}), R^i(1^\gamma, \{b_i\}))$ where L^i, R^i abbreviate L^{g_i}, R^{g_i} respectively. It is easy to see, however, that the structures M_i are all identical. □

By Claim 3 and by Theorem 2 in §5.2 we have that p_1 forces $T \models \phi^*(U, a_1)$ and p_2 forces $T \models \phi^*(U, a_2)$. Then $p_1 \cup p_2$ forces both these statements, which contradicts our assumption about ϕ^*. The Main Theorem is proved. □

REFERENCES

J.R. BÜCHI [1962], *On a decision method in second order arithmetic*, ***Logic, methodology and philosophy of science (Proceedings of the 1960 international congress)***, Stanford University Press, Stanford, California, pp. 1–11.

J.R. BÜCHI and L.H. LANDWEBER [1969], *Solving sequential conditions by finite-state strategies*, ***Transactions of the American Mathematical Society***, vol. 138, pp. 295–311.

Y. GUREVICH [1979], *Modest theory of short chains*. I, this JOURNAL, vol. 44, pp. 481–490.

Y. GUREVICH and L.A. HARRINGTON [1982], *Automata, trees, and games*, ***Proceedings of the Fourteenth ACM Symposium on the Theory of Computing*** (San Francisco, May, 1982) (to appear).

H. LÄUCHLI [1968], *A decision procedure for the weak second order theory of linear order*, ***Contributions to mathematical logic (Proceedings of the logic colloquium, Hannover, 1966)***, North-Holland, Amsterdam, pp. 189–197.

M.O. RABIN [1969], *Decidability of second-order theories and automata on infinite trees*, ***Transactions of the American Mathematical Society***, vol. 141, pp. 1–35.

——— [1972], ***Automata on infinite objects and Church's problem***, Conference Board of the Mathematical Sciences Regional Conference Series in Mathematics, no. 13, American Mathematical Society, Providence, R.I.

S. SHELAH [1975], *The monadic theory of order*, ***Annals of Mathematics***, ser. 2, vol. 102, pp. 379–419.

J.R. SHOENFIELD [1971], *Unramified forcing*, ***Axiomatic set theory***, Proceedings of Symposia in Pure Mathematics, vol. 13, part 1, American Mathematical Society, Providence, R.I., pp. 357–381.

UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN 48109

HEBREW UNIVERSITY
JERUSALEM, ISRAEL