FIXED POINTS vs. INFINITE GENERATION

Damian Niwiński

Institute of Mathematics, University of Warsaw
PKiN IXp. Pl 00 901 Warsaw

Abstract: We characterize Rabin definability of properties of infinite trees by means of fixed point definitions composed from the basic operations of a standard powerset algebra of trees and involving the *least and greatest* fixed point operators besides the finite union operator and functional composition. Also, a strict connection is established between a hierarchy resulted from alternating the least and greatest fixed point operators and the hierarchy induced by *Rabin indicies* of automata. The characterization result is actually proved on a more general level, namely for arbitrary *powerset algebra*, where the concept of Rabin automaton is replaced by a more general concept of infinitary grammar.

Introduction

Several autors established recently elementary upper bounds for decision problems for a variety of propositional modal logics of programs *via* reduction to emptiness problem of Rabin automata on infinite trees (Streett [19]Vardi & Wolper [22], Vardi & Stockmeyer [21], Danecki [6]). In doing this, they followed the earlier ideas of Büchi [3] and Rabin [15] who had originally invented automata on infinite objects in context of decision problems in logic. The purpose of automata is there to check validity of logical formulae in the structures actually presented as infinite words or trees. In context of program logics this means that, roughly speaking, once a computation of a program is modelled by a (possibly infinite) tree, Rabin automata are capable of testing this tree for properties expessible in these logics like terminating, looping, repeating, fairness, etc.

On the other hand, the inductive nature of most of interesting properties of programs follows that one can accurately characterize them by using the least and the greatest fixed point operators, the latter applied to capturing infinite computations (Emerson & Clarke [7], Park [13], Kozen [9].

Our aim here is to show a kind of equivalence between these two modes of expression of properties of infinite computations: automata and fixed points. We consider a powerset algebra of trees whose basic operations are inherited from the standard tree algebra in a usual way. A fixed point definition is built from the basic operations and projections by means of the following operators: set-union, functional composition, the least and the greatest fixed point operators μ and ν. We show that a set of trees is fixed point definable iff it is definable by a Rabin automaton. Also, a strict connection can be drawn between a hierarchy of tree languages resulted from increasing the number of alternations between μ and ν and the hierarchy induced by increasing the index of automaton (I showed in [11] and [12] that the both hierarchies are infinite).

In fact, we proved the above-mentioned characterization on a more general level, namely for an arbitrary powerset algebra (as considered, e.g. by Courcelle [5]). This of course requires a generalization of a Rabin automaton into a device which could "run" over an arbitrary algebraic structure; it proceeds in a natural way. However in this new context, we prefer to call the device in question a *grammar* rather than an automaton and view its action as *generating* rather than "testing". (In terms of our general considerations above, one can generate models with certain property rather than test all models for that property.)

A series of related previous papers could be mentioned, going back to the classical characterization of (finitary) context-free languages by means of the least solutions of equations in a powerset algebra of finite words, due to Ginsburg & Rice [8] and Schützenberger [18]. The greatest fixed points as a mode of representing non-terminating computations appeared in Arnold & Nivat [1] in context of semantics of nondeterministic recursive programs and in Park [13] in context of semantics of fair parallelism. Park [14] provides a complete characterization of regular sets of finite/infinite words by means of the both extremal fixed point operators. A similar result for ω-context-free languages was shown in [10]. Notably, in the both cases, the above mentioned hierarchy of alternations between μ and ν turns out to coincide on the $\nu\mu$-level. The analogous level of the hierarchy in the powerset algebra of trees has been shown to coincide with definability by Büchi automata on infinite trees (Takahashi [20], Niwiński [11]). Arnold & Niwiński [2] proved that this characterization continues to hold when the intersection operator is incorporated; the question if the hierarchy in *that* case is infinite, remains open.

Vector notation. Throughout the paper we shall often abbreviate a vector $\alpha_1,\dots,\alpha_n$ by $\mathbf{a}$, also in more complex terms. So we write, e.g. $\mathbf{a} \leq \mathbf{b}$ instead of $\alpha_1 \leq \beta_1$ & ...& $\alpha_n \leq \beta_n$ or even $\mathbf{X} \subseteq \mathbf{Y}$ for $X_1 \subseteq Y_1$ &...& $X_m \subseteq Y_m$; also sometimes $\mathbf{G}(\mathbf{x})$ for $G(x_1,\dots),\dots$ etc.

1 Trees

For a set X, X^* denotes the set of finite words over X including the empty word λ. We write $\leq$ ($<$) for the (proper) initial segment relation. A subset $\{1,\dots,n\}$ of the set of natural numbers ω is abbreviated [n].

Given a set Σ, a *Σ-valued tree* is a mapping $t : \mathrm{Dom}\ t \to \Sigma$, where Dom t is a non-empty subset of ω^* closed under initial segments. For a node w in Dom t, the subtree t_w of t is a tree with $\mathrm{Dom}\ t_w = \{ v : wv \in \mathrm{Dom}\ t \}$ defined by $t_w(v) = t(wv)$. If w and wi are in Dom t ($i\in\omega$), wi is called a *childe* of w. Nodes without children are called *leaves*. A *path* is an infinite sequence $P: w_0, w_1,\dots$ such that each w_{n+1} is a child of w_n. In this context we also set:

$$\mathrm{Inf}(P) = \{\alpha\in\Sigma : t(w_n)=\alpha \text{ for infinitely many } n \}.$$

Let *Sig* be a finite signature, viz. a set of function symbols, each f in *Sig* given with an arity $ar(f)\geq 0$. A *syntactic tree* over *Sig* is a *Sig*-valued tree t such that for any w in Dom t, if the arity of t(w) is k then the children of w are exactly w1,...,wk. Note that the only leaves of a syntactic tree are the nodes labelled by constant symbols. Let Tree(*Sig*) denote the set of syntactic trees over *Sig*.

2 Powerset algebras

Suppose $\mathbf{A}$ is an algebra over *Sig*,

$$\mathbf{A} = \langle A; f^{\mathbf{A}} : f \in Sig \rangle,$$

where $f^{\mathbf{A}} : A^{ar(f)} \to A$ is an interpretation of f. By the powerset algebra of $\mathbf{A}$, we mean a system

$$\mathbf{PA} = \langle \mathcal{P}(A); f^{\mathbf{PA}} : f \in Sig \rangle,$$

whose universe is the powerset of A and the interpretation of $f \in Sig$, say $ar(f) = k$, is defined by

$$f^{\mathbf{PA}}(S_1,\dots,S_k) = \{f^{\mathbf{A}}(s_1,\dots,s_k) : s_i\in S_i\}.$$

Note that $\mathcal{P}(A)$ is completely ordered by the subset relation $\subseteq$ and the basic operations $f^{\mathbf{PA}}$ are monotonic in all variables. We mention a few examples which will be revisited in the sequel.

2.1.*Example*. Let

$$TSig = \langle \mathrm{Tree}(Sig); f^{TSig} : f \in Sig \rangle,$$

where, for each f in *Sig*, say $ar(f)=k$, and $t_1,\dots,t_k$ in Tree(*Sig*), $f^{TSig}(t_1,\dots,t_k)$ is the only tree t such that $t(\lambda) = f$ and the subtree of t induced by the node $\langle i\rangle$, $t_{\langle i\rangle}$, is just t_i, for $i \in [k]$. We call the system **PT***Sig* *a powerset algebra of trees* (over *Sig*).□

2.2.*Example*. Let $\Sigma = \{\sigma_1,\dots,\sigma_m\}$ be a finite alphabet and let $\Sigma^\alpha = \Sigma^* \cup \Sigma^\omega$ where Σ^ω is the set of all ω-words over Σ. Consider an algebra

$$\mathbf{CF}\Sigma = \langle \Sigma^\alpha; \cdot, \sigma_1,\dots,\sigma_m, \lambda \rangle$$

where $\cdot$ is the operation of concatenation which is extended to Σ^α by setting, for an infinite $u = u_0u_1\dots$,

$$u\cdot v = u, \text{ for any } v,$$

$$w_0\dots w_k\cdot u = w_0\dots w_k u_0 u_1\dots$$

Let

$$\mathbf{R}\Sigma = \langle \Sigma^\alpha, \sigma_1\cdot,\dots,\sigma_m\cdot, \lambda \rangle$$

be a system obtained from the previous one by restricting the concatenation to the left multiplication by single letters.

The corresponding powerset algebras may be viewed as two variants of a powerset algebra of finite/infinite words (the denotations comming from *context-free* and *regular* respectively).□

3 μ-Terms and their semantics

We fix an infinite list of variables *Var*, its elements will be usually denoted $x, y, z, \ldots, x_i, \ldots$ etc. The set of μ-terms over *Sig*, *μ-Term Sig*, is defined by the following clauses:

- the variables are in *μ-Term Sig*,
- if $f \in Sig$, say $ar(f)=k$, $t_1, \ldots, t_k \in$ *μ-Term Sig* then $f(t_1, \ldots, t_k) \in$ *μ-Term Sig*
- if $t_1, t_2 \in$ *μ-Term Sig* then $(t_1 \cup t_2) \in$ *μ-Term Sig*,
- if $t \in$ *μ-Term Sig* and x is a variable then $\mu x.t$, $\nu x.t \in$ *μ-Term Sig* .

In writing $t = t(x_1, \ldots, x_k)$ (or $t(x)$ for short), we indicate that the free variables of t (viz. not bound by μ or ν) are from among $x_1, \ldots, x_k$. We denote by $t[t_1/x_1, \ldots, t_k/x_k]$, or $t[\mathbf{t}/\mathbf{x}]$ for short, the result of simultaneous substituting the μ-terms $t_1, \ldots, t_k$ for all free occurrences of the variables $x_1, \ldots, x_k$ in t respectively and assume that before the substitution allbound variables of t which occur in the t_i's are renamed in some proper way.

Let **A** be an algebra over *Sig*. An *interpretation of a μ-term $t(x_1, \ldots, x_n)$ in the powerset algebra* **PA** , under a *valuation* of x_i by S_i, $S_i \subseteq A$ for $i \in [n]$, in symbols $t^{PA}[x_1 : S_1, \ldots, x_n : S_n]$ or $t^{PA}[S_1, \ldots, S_n]$ for short, is defined inductively as follows:

$$x_i^{PA}[S] = S_i ,$$

$$f(t_1, \ldots, t_k)^{PA}[S] = f^{PA}(\mathbf{t}^{PA}[S]) ,$$

$$(t_1 \cup t_2)^{PA}[S] = t_1^{PA}[S] \cup t_2^{PA}[S],$$

$\mu y.t^{PA}[S]$, resp. $\nu y.t^{PA}[S]$, is the *least*, resp. the *greatest*, solution of the equation

$$x = t^{PA}[y : x,\ x : S]$$

The correctness of the above definition is based on the following

Knaster-Tarski Fixed point theorem *A monotonic mapping $f : L \to L$ of a complete lattice $\langle L, \leq \rangle$ has the least fixed point*

$$\mu x.f(x) = \bigcap \{ u \in L : f(u) \leq u \}$$

and its dual, the greatest one

$$\nu x.f(x) = \bigcup \{ u \in L : u \leq f(u) \} .\square$$

3.1. *Example*. Let $\Sigma = \{0,1\}$ and consider the powerset algebra **PR**Σ of Example 2.2. One can verify

$$(\mu x.(0 \cdot x \cup 1))^{PR\Sigma} = 0^*1,$$

$$(\nu x.01x)^{PR\Sigma} = 010101\ldots,$$

$$(\nu y.\mu x.(0x \cup 1x \cup 01y))^{PR\Sigma} = (\{0,1\}^*01)^\omega$$

□

From now on we assume that, unless otherwise stated, we work with a fixed signature and therefore we shall omit the suffix *Sig* in notation concerning μ-terms etc.

For a set $M \subseteq$ *μ-Term*, let μM be the least set of μ-terms such that

- $M \subseteq \mu M$,
- if $t, t_1, \ldots, t_k \in \mu M$ then $t[\mathbf{t}/\mathbf{x}] \in \mu M$,
- if $t \in \mu M$ then $\mu x.t \in \mu M$.

Let νM be defined analogously, with μ replaced by ν.

The following variant of the well-known *Bekic Principle* will be useful in the sequel.

3.2. Lemma. *Suppose $t_1(x_1, \ldots, x_k, z_1, \ldots, z_m)$, $\ldots, t_k(\mathbf{x}, \mathbf{z}) \in M \subseteq$ μ-Term. Then there exist $s_1(\mathbf{z}), \ldots, s_k(\mathbf{z})$ in μM such that in any powerset algebra* **PA** *and for any $R_1, \ldots, R_m \subseteq A$, the tuple $s_1^{PA}[\mathbf{z} : \mathbf{R}], \ldots, s_k^{PA}[\mathbf{z} : \mathbf{R}]$ is the least solution of the system of equations*

$$\xi_1 = t_1^{PA}[\mathbf{x} : \xi,\ \mathbf{z} : \mathbf{R}],$$

$$\ldots\ldots\ldots\ldots\ldots\ldots$$

$$\xi_k = t_k^{PA}[\mathbf{x} : \xi,\ \mathbf{z} : \mathbf{R}] .$$

The similar for ν and the greatest solution.□

4 Fixed point hierarchy

Let *μ-Term$_0$* be the set of μ-terms without any occurrences of μ ,ν . The set of all μ-terms may be naturally organized into a hierarchy of classes *μTerm$_0$*, *νTerm$_0$*, *$\nu\mu$Term$_0$*, *$\mu\nu$Term$_0$* etc. Rather than to follow the notation of arithmetical hierarchy (as I actually did in [11]), we prefer now a "zigzag" denotation of these classes:

$$Nu_0 = \mu Term_0 , \qquad Mu_0 = \nu Term_0 ,$$

$$Nu_{n+1} = \mu\nu\ Nu_n , \qquad Mu_{n+1} = \nu\mu\ Mu_n .$$

Intuitively, in Nu_n (Mu_n) n refers to the "essential ν-depth (μ-depth)" of a μ-term. Note that, by definition, $Mu_n \cup Nu_n \subseteq Mu_{n+1} \cap Nu_{n+1}$ for $n < \omega$ and $\bigcup_{n<\omega} Nu_n = \bigcup_{n<\omega} Mu_n = \mu\text{-}Term$.

Given a powerset algebra **PA** and a *closed* μ-term t, the interpretation $t^{\mathbf{PA}}$ [], say $t^{\mathbf{PA}}$ for short, is a subset of A. For a set $M \subseteq \mu\text{-}Term$ let

$$M(\mathbf{PA}) = \{ t^{\mathbf{PA}} : t \text{ is a closed } \mu\text{-term in } M \}$$

4.1.*Example*. Park [14] shows that for the algebra **PRΣ** of Example 2.2

$$\bigcup_{n<\omega} Nu_n(\mathbf{PR\Sigma}) = \nu\mu\ Term_0(\mathbf{PR\Sigma}) .$$

For the algebra **PCFΣ**, I prove [12]

$$\bigcup_{n<\omega} Nu_n(\mathbf{PCF\Sigma}) = \mu\nu Term_0(\mathbf{PCF\Sigma}) \cap \nu\mu Term_0(\mathbf{PCF\Sigma}). \square$$

4.2.*Example*. Let, for $n<\omega$, $Sig_n = \{a_0, \ldots, a_n\}$ be a signature where each a_i is of arity 2. Consider a sequence of μ-terms

$$t_0 = \nu x_0 . a_0(x_0, x_0),$$

$$t_1 = \mu x_1 . \nu x_0 . (a_0(x_0, x_0) \cup a_1(x_1, x_1)),$$

$$t_2 = \nu x_2 . \mu x_1 . \nu x_0 . (a_0(x_0, x_0) \cup a_0(x_1, x_1) \cup$$

$$\cup\ a_2(x_2, x_2)),$$

....

and let a sequence s_n be defined dually, with μ and ν interchanged. Let

$$\mathbf{M}_n = t_n^{\mathbf{PT}Sig_n} , \quad \mathbf{N}_n = s_n^{\mathbf{PT}Sig_n}$$

(c.f. Example 2.1).

In [11] (and in more details in [12]) I showed that

$$\mathbf{M}_{2n} \in Mu_n(\mathbf{PT}Sig_{2n}) - Nu_n(\mathbf{PT}Sig_{2n}),$$

$$\mathbf{M}_{2n+1} \in Nu_{n+1}(\mathbf{PT}Sig_{2n+1}) - Nu_n(\mathbf{PT}Sig_{2n+1})$$

and the $\mathbf{N}_n$'s satisfy the dual conditions. Also, all the family can be encoded over a single alphabet, e.g. Sig_1, proving that the hierarchy of classes Mu_n, Nu_n in $\mathbf{PT}Sig_1$ is strict.

The above sets have an interesting meaning: for $n<\omega$, $\mathbf{M}_n$ is the set of all trees in $TSig_n$ possessing the following property:

-for any path P, if $a_i \in \mathrm{Inf}(P)$ and i is *odd* then there exists $j > i$, j ***even***, such that $a_j \in \mathrm{Inf}(P)$.

In particular, $\mathbf{M}_1$ is the set of all binary trees labelled by $\{a_0, a_1\}$ such that on each path there is only finitely many a_1 . The family $\mathbf{N}_n$ has a similar characterization.$\square$

5 Grammars

A ***grammar*** over a signature Sig is a tuple

$$G = \langle V, V_T, x_0, Tr, Acc \rangle ,$$

where V is a finite set of variables, $V_T \subseteq V$ are terminal or ***free variables*** of G, $x_0 \in V - V_T$ *is the start symbol*, Tr is a set of *transitions* and Acc is an *acceptance condition*. Set $V_N = V - V_T$. Each α in Tr has one of the following forms:

(1) $\alpha : x \to f(y_1, \ldots, y_k)$,
where $x \in V_N, y_1, \ldots, y_k \in V$, $f \in Sig$,

(2) $\alpha : x \to y$,
where $x \in V_N$, $y \in V$,

(3) $\alpha : z \to$,
where $z \in V_T$.

To a transition α, we asssociate concepts of the *arity of* α, $\mathrm{ar}(\alpha)$, the *input-variable of* α, $in\text{-}var(\alpha)$, and, for i $\in [\mathrm{ar}(\alpha)]$, the *i-th output-variable of* α, $i\text{-}out\text{-}var(\alpha)$, defined as follows.

In the case (1) :

$\mathrm{ar}(\alpha) = k$,

$in\text{-}var(\alpha) = x$,

$i\text{-}out\text{-}var(\alpha) = y_i$;

in the case (2) :

$\mathrm{ar}(\alpha) = 1$,

$in\text{-}var(\alpha) = x$,

$1\text{-}out\text{-}var(\alpha) = y$;

in the case (3) :

$\mathrm{ar}(\alpha) = 0$,

$in\text{-}var(\alpha) = z$.

We shall denote by $Tr(x)$ the set of transitions whose input variable is x .

The acceptance condition has the form

$\langle (U_1,L_1),\ldots,(U_n,L_n) \rangle$,

where $U_i, L_i \subseteq V_N$ for $i \in [n]$. The number n is called the *index* of G and denoted $\mathrm{ind}(G)$.

A *derivation* in G is any Tr-valued tree d satisfying the following conditions :

(a) $in\text{-}var(d(\lambda)) = x_0$.

(b) For any $w \in \mathrm{Dom}\ d$, if the arity of $d(w)$ is k then w has exactly k children : $w1,\ldots,wk$ and $in\text{-}var\ (d(wi)) = i\text{-}out\text{-}var(d(w))$, for $i \in [k]$.

(c) Suppose P: $w_0, w_1, \ldots$ is an infinite path in d and let

$\mathrm{Inf\ Var}\ (P) = \{in\text{-}var(\alpha) : \alpha \in \mathrm{Inf}(P)\}$.

Then there exists $j \in [n]$ such that

$\mathrm{Inf\ Var}(P) \cap U_j \neq \phi$ and

$\mathrm{Inf\ Var}(P) \cap L_j = \phi$.

Now let **A** be an algebra over *Sig*.Suppose $V_T = \{z_1,\ldots,z_m\}$ and $S_1,\ldots,S_m \subseteq A$. A *realization of a transition* $\alpha \in Tr$ in **PA** under a valuation $z_1:S_1,\ldots,z_m:S_m$, is a sequent of the form β: $a \to v$, where $a \in A$, $w \in A^*$, $\mathrm{length}(w) = \mathrm{ar}(\alpha)$ and

- if $\alpha : x \to f(x_1,\ldots,x_k)$ then

$\beta : a \to a_1 \ldots a_k$, where

$a = f^{A}(a_1,\ldots,a_k)$;

-if $\alpha : x \to y$ then

$\beta : a \to a$;

-if $\alpha : z_i \to$ then

$\beta : b \to \lambda$, where $b \in S_i$

A *realization of a derivation d of G*, in **PA**, under a valuation $z:S$ is a tree

$r : \mathrm{Dom}\ r \to A$

with $\mathrm{Dom}\ r = \mathrm{Dom}\ d$, such that, for any $w \in \mathrm{Dom}\ r$, if w has exactly k children ($k \geq 0$) $w1,\ldots wk$ then

$r(w) \to r(w1) \ldots r(wk)$

is a realization of $d(w)$ in **PA** under $z:S$.

For an element $a \in A$, an *expansion of* a by G under a valuation $z:S$ is a pair (d,r) where d and r are as above and moreover $r(\lambda) = a$.

As for terms, we shall write $G = G(z_1,\ldots, z_m)$ to indicate that the free variables of the grammar G are from among z .The *interpretation of* $G(z)$ *in* **PA** under a valuation $z:S$ is defined by

$$G^{PA}\ [z:S] = \{a \in A : \text{there exists an expansion of a by } G \text{ in } \mathbf{PA} \text{ under the valuation } z:S\ \}$$

5.1.*Example*. Let $\Sigma = \{a_1,\ldots,a_l\}$ be an alphabet and let $T\Sigma$ be the set of all trees $t : [2]^* \to \Sigma$. A *Rabin automaton* [17] over Σ is a tuple $\mathcal{A} = (Q, q_0, \delta, \mathrm{Acc})$, where Q is a finite set of states, $q_0 \in Q$, $\delta : Q \times \Sigma \to \mathcal{P}(Q \times Q)$, and $\mathrm{Acc} = \langle (U_1,L_1),\ldots,(U_n,L_n)\rangle$. The number n is the index of the automaton. A run of $\mathcal{A}$ on a tree t is a Q-valued tree r with $\mathrm{Dom}\ r = \mathrm{Dom}\ t$ such that $r(\lambda) = q_0$ and $(r(w1),r(w2)) \in \delta(r(w))$ for $w \in \mathrm{Dom}\ r$. $\mathcal{A}$ accepts t if there is a run such that, for each path $P: w_0, w_1, \ldots$, there is $i \in [n]$, such that $\mathrm{Inf}(P) \cap U_i \neq \phi$ and $\mathrm{Inf}(P) \cap L_i = \phi$. Let $T(\mathcal{A})$ be the set of trees accepted by $\mathcal{A}$.

Now consider Σ as a signature, where each a_i is of arity 2. Given an automaton $\mathcal{A}$ as above, let $G_{\mathcal{A}}$ be a grammar over Σ defined by $V = Q$, $V_T = \phi$, $x_0 = q_0$, $Tr = \{ q \to a(q',q'') : (q',q'') \in \delta(q,a) \}$, $Acc = \mathrm{Acc}$. It is easy to check that if (d,r) is an expansion of t by $G_{\mathcal{A}}$ in $\mathbf{PT\Sigma}$ then a tree r' with $\mathrm{Dom}\ r' = \mathrm{Dom}\ r = [2]^*$, defined by $r'(w) = in\text{-}var(d(w))$ is an accepting run of $\mathcal{A}$ on t (note that $r(w) = t_w$ for $w \in \mathrm{Dom}\ r$).

On the other hand, it follows easily from the *Rabin tree theorem* that a tree language definable by a grammar is also definable by a Rabin automaton; with some care, one can choose an automaton with the same index.□

5.2.*Example*. For the algebra $\mathbf{PCF\Sigma}$ of Example 2.2, one can show that our grammars may be (with some care) transformed into the *α-context-free grammars* as considered by Cohen & Gold [4] and Niwiński [10] and *vice versa*. The similar for $\mathbf{PR\Sigma}$ and α-regular grammars.□

6 Characterization result

We say that a grammar $G(z_1,\ldots,z_m)$ is *equivalent* to a μ-term $t(z_1,\ldots,z_m)$ if for any powerset algebra **PA** and $S_1,\ldots,S_m \subseteq A$,

$$t^{PA}[z:S] = G^{PA}[z:S] \ .$$

6.1. Theorem. (I) *For any μ-term $t(z_1,\dots,z_m)$, there exists a grammar $G(z_1,\dots,z_m)$, equivalent to t. Moreover, if t is in Nu_n then G may be chosen with index* n.

(II) *For any grammar $G(z_1,\dots,z_m)$, there is a μ-term $t(z_1,\dots,z_m)$, equivalent to G. Moreover, if* ind(G) = n, *then t may be chosen in Nu_n.*

Sketch of proof. Part I has been shown in [11] for the case where the notion of equivalence was restricted to **PTSig**. The proof for the general case is similar, it is contained in [12] and will be available in the full version of the paper. In the sequel we sketch the argument for Part II.

The following operations on grammars will be needed. Let G be as above.

(1) Suppose $L \subseteq V_N$, $x_0 \not\in L$. Then G^{freeL} is a grammar obtained from G by "giving freedom" to variables in L:

$$G^{freeL} = \langle V', V_T', x_0', Tr', Acc' \rangle,$$

where

$V' = V$,

$V_T' = V_T \cup L$,

$x_0' = x_0$,

$Tr' = (Tr - \{y \to t : y \in L\}) \cup \cup \{y \to : y \in L\}$

$Acc' = \langle (U_1', L_1'),\dots,(U_n', L_n')\rangle$

where $U_i' = U_i - L$, $L_i' = L_i - L$ for $i \in [n]$.

(2) Suppose $x \in V_N$. The G^{startx} is obtained from G as follows:

$$G^{startx} = \langle V'', V_T'', x_0'', Tr'', Acc''\rangle,$$

where x_0'' is a new variable, $V'' = V \cup \{x_0''\}$, $Acc'' = Acc$ and

$Tr'' = Tr \cup \{x_0'' \to t : (x \to t) \in Tr\}$.

Now the proof proceeds by induction on the index of G. If ind(G) = 0 then no infinite paths in derivations in G are allowed. Then the set Tr may be viewed as a system of equations and it is standard to prove that the least solution of this system corresponds to a tuple of sets finitary generated by the grammar (c.f. Courcelle [5]).

Now suppose the thesis holds for all grammars with the index $\leq$ n. The proof proceeds in two steps.

Step 1. We consider a grammar $G = \langle V, V_T, x_0, Tr, Acc\rangle$ with the acceptance condition in the form:

$Acc = \langle (U_1, L_1),\dots,(U_n, L_n), (U, \phi)\rangle$.

We may assume $U \neq \phi$, $x_0 \not\in U$, say $U = \{x_0,\dots,x_k\}$. Consider the grammar G^{freeU} and, for $i \in [k]$, the grammar $(G^{startx_i})^{freeU}$. Note that the indices of these grammars are $\leq$ n (the pair (ϕ,ϕ) may be deleted from the acceptance condition) and then, by the induction hypothesis, we have the corresponding μ-terms in Nu_n, say $t(x,z)$, $t_1(x,z),\dots, t_k(x,z)$ (where $V_T = \{z\}$). Now consider a system of equations

$x_1 = t_1$,

....

$x_k = t_k$.

By Lemma 3.2, there is a vector of μ-terms in νNu_n, $s_1(z),\dots,s_k(z)$, which represents the greatest solution of this system in any powerset algebra. We claim:

(1) G^{startx_i} is equivalent to s_i, $i \in [k]$;

(2) G is equivalent to $t[s_1/x_1,\dots,s_k/x_k]$.

The clause (2) follows easily from (1). To prove (1) consider a powerset algebra **PA** and let $S_1,\dots,S_m \subseteq A$. Let us abbreviate G^{startx_i} by G_i. By Knaster-Tarski theorem, it is enough to show two facts:

(i) $\forall$ **M** : $\mathbf{M} \subseteq t^{\mathbf{PA}}[\mathbf{M},\mathbf{K}]$ implies $\mathbf{M} \subseteq G^{\mathbf{PA}}[\mathbf{K}]$;

(ii) $G^{\mathbf{PA}}[\mathbf{K}] \subseteq t^{\mathbf{PA}}[G^{\mathbf{PA}}[\mathbf{K}], \mathbf{K}]$.

Ad(i). Suppose $a \in M_i$. Then $a \in t_i^{\mathbf{PA}}[\mathbf{M},\mathbf{K}]$ and, by choice of t_i, $a \in (G^{startx_i})^{freeU}[\mathbf{M},\mathbf{K}]$. Consider an expansion of a, say (d, r). For each leaf wof d such that $in\text{-}var(d(w)) = x_j$, we have $r(w) \in M_j$. Then again, there is an expansion of this r(w) by $(G^{startx_j})^{freeU}$ and so on. By combining the expansions produced recursively in that way, we eventually obtain a desired expansion of a by G_i.

Ad(ii). Suppose $a \in G_i^{PA}[K]$ and let (d,r) be an expansion of a. Let

$E = \{w \in \text{Dom } d: \textit{in-var}(d(w)) \in U$ and, for $v < w$, $\textit{in-var}(d(v)) \notin U\}$

The restriction of (d,r) to the set

$E' = \{v \in \text{Dom } d :$ there is no $w \in E$ such that $w < v\}$

can be viewed, after an obvious modification as an expansion of a by $(G^{startx}{}_i)^{freeU}$ under the valuation $x : G^{PA}[K]$, $z : K$, proving that $a \in t_i^{PA}[G^{PA}[K], K]$.

Step 2. Consider a grammar $G = \langle V, V_T, x_0, Tr, Acc\rangle$ with the acceptance condition $Acc = \langle(U_1,L_1),\dots,(U_n,L_n),(U_{n+1},L_{n+1})\rangle$. Let $L = L_1 \cup \dots \cup L_{n+1}$, say $L = \{x_1,\dots,x_p\}$, $x_0 \notin L$. Let, for $i \in [n+1]$,

$G_i' = G^{freeL}{}_i$,

and, for $x \in L$,

$G_{i,x} = (G^{startx})^{freeL}{}_i$.

Note that these grammars have the acceptance conditions in the form considered already in *Step* 1. Therefore, there exist corresponding μ-terms in νNu_n, say t_i', $t_{i,x}$. Let

$t' = t_1' \cup \dots \cup t_{n+1}'$

$(t' = t'(x_1,\dots,x_p,z))$,

$t_x = t_{1,x} \cup \dots \cup t_{n+1,x}$.

Consider a system of equations

$x_1 = t_{x_1}$,

$x_p = t_{x_p}$.

By Lemma 3.2, there are some μ-terms in $\mu\nu(Nu_n) = Nu_{n+1}$, say $s_1,\dots,s_p$, which represent the least solution of this system. We claim:

(1) $G^{startx}{}_i$ is equivalent to s_i, $i \in [p]$;

(2) G is equivalent to $t'[s_1/x_1,\dots,s_p/x_p]$.

(Note that the last term is in Nu_{n+1}). The clause (2) follows easily from (1). To prove (1), consider a powerset algebra **A** and let $S_1,\dots,S_m \subseteq$ A. Let us abbreviate t_{x_i} by t_i and $G^{startx}{}_i$ by G_i. By Knaster-Tarski theorem, it is enough to show two facts:

(i) $\forall M : t^{PA}[M,K] \subseteq M$ implies $G^{PA}[K] \subseteq M$;

(ii) $t^{PA}[G^{PA}[K], K] \subseteq G^{PA}[K]$.

We give the idea of an argument for (i); the proof of (ii) is similar as in the previous step. Suppose $a \in G_i^{PA}[K]$ and let (d,r) be an expansion of a. Define a sequence of subsets $E_l \subseteq$ Dom d and a sequence of integers $q_l \in [n+1]$ as follows:

$E_0 = \{\lambda\}$,

$q_0 = 1$.

Now suppose that E_l, q_l are defined. For each $w \in E_l$, let

$D_w = \{v \in \text{Dom } d : w < v,\ \textit{in-var}(d(v)) \in L_{q_l}$, and for any v', $w < v' < v$, $\textit{in-var}(d(v')) \notin L_{q_l}\}$.

Set

$E_{l+1} = \bigcup_{w \in E_l} D_w$,

$q_{l+1} = q_l + 1$ if $q_l < n+1$,

$= 1$ otherwise.

Set

$E = \bigcup_l E_l$.

It follows from definition of derivation that E contains no infinite chains. We claim that for any $w \in E$, if $\textit{in-var}(d(w)) = x_j$ then $r(w) \in M_j$, in particular $a = r(\lambda) \in M_i$ as required. This follows inductively from well-foundness of E and the fact that if all successors in E of some node $w \in E$ have the desired property then w has also this property.

This remark completes the proof.□

References

[1] A.Arnold & M.Nivat, Non deterministic recursive programs, *in* "Fundamentals of Computation Theory", pp.12-21, Lectures Notes in Computer Sci. No.56, 1977.

[2] A.Arnold & D.Niwiński, Fixed points characterization of Büchi automata on infinite trees, Rep. I-8730, Université de Bordeaux 1987.

[3] J.R.Büchi, On a decision method in restricted second order arithmetic, *in* "Logic, Metodology and Philosophy of Science. Proc. 1960 Intern.Congr." (E.Nagel *et al.* Eds.), Stanford Univ. Press, pp.1-11.

[4] R.Cohen & A.Gold, Theory of ω-languages, I,II, *J.Comput. System Sci.* **15**, 1977, *pp.*169-208.

[5] B.Courcelle, Equivalence and transformations of regular systems. Applications to recursive program schemes. *Theor.Comput.Sci.* **42**, 1986, pp.1-122.

[6] R.Danecki, Propositional dynamic logic with strong loop predicate, *in* "Mathematical Foundations of Computer Science. Proc.Int.Symp. 1984", LNCS 176, pp.573-581.

[7] A.E.Emerson & E.C.Clarke, Characterizing Correctness Properties of Parallel Programs using Fixpoints, *in* Proc.7th Int.Coll.on Automata, Languages and Programming, 1980,pp.169-181.

[8] S.Ginsburg & H.G.Rice, Two families of languages related to ALGOL, *J.ACM* **9**,1962 pp.350-371.

[9] D.Kozen, Results on the propositional μ-calculus, *Theor.Comput.Sci.* **27**, 1983, pp.333-354.

[10] D.Niwiński, Fixed point characterization of context-free α-languages, *Inform.Contr.* **63**, 1984, pp. 247-276.

[11] D.Niwiński, On fixed point clones, Proc. 13th.Int.Coll. on Automata, Languages and Programming 1986, pp.464-473.

[12] D.Niwiński, A hierarchy of objects definable in fixed point calculus (in Polish), Ph.D. thesis, University of Warsaw 1987.

[13] D.Park, On the semantics of fair parallelism, *in* "Abstract software specifications" LNCS 86, 1980, pp.504-526.

[14] D.Park, Concurrency and automata on infinite sequences, *in*: Theoretical Computer Science (P.Deussen, Ed.), LNCS 104, 1981, pp.167-183.

[15] M.O.Rabin, Decidability of second - order theories and automata on infinite trees, *Trans.Amer.Math.Soc.* **141**, 1969, pp.1-35.

[16] M.O.Rabin, Weakly definable relations and special automata, *in* Math.Logic and Found. of Set Theory (Bar-Hillel, Ed.), North Holland 1970, pp.1-23.

[17] M.O.Rabin, Automata on infinite objects and Church's problem, *Proc.Regional AMS Conf.Series in Math.* **13**, 1972, pp.1-22.

[18] P.M.Schützenber, Push-down automata and context-free languages, *Inform. Contr.* **6**, 1963, pp.246-264.

[19] R.S.Streett, Propositional Dynamic Logic of Looping and Converse is elementarily decidable, *Inform.Contr.* **54**, 1982, pp.121-141.

[20] M.Takahashi, The greatest fixed points and rational omega-tree languages *Theor.Comput.Sci.* **44**, pp.259-274.

[21] M.Y.Vardi & L.Stockmeyer, Improved upper and lower bounds for modal logics of programs, *Proc. 17th ACM STOC*, 1985.

[22] M.Y.Vardi & P.Wolper, Automata - theoretic technics for modal logics of programs, *Proc. 16th ACM STOC*, 1984, pp. 446-456.