Theoretical Computer Science 25 (1983) 95-169
North-Holland Publishing Company 

# FUNDAMENTAL PROPERTIES OF INFINITE TREES

Bruno COURCELLE
*UER de Mathématiques et Informatique, Université Bordeaux-I, 33405 Talence, France*

Communicated by M. Nivat
Received July 1981
Revised March 1982

To the memory of C.C. Elgot

**Abstract.** Infinite trees naturally arise in the formalization and the study of the semantics of programming languages. This paper investigates some of their combinatorial and algebraic properties that are especially relevant to semantics.

This paper is concerned in particular with regular and algebraic infinite trees, *not* with regular or algebraic *sets* of infinite trees. For this reason most of the properties stated in this work become trivial when restricted either to finite trees or to infinite words.

It presents a synthesis of various aspects of infinite trees, investigated by different authors in different contexts and hopes to be a unifying step towards a theory of infinite trees that could take place near the theory of formal languages and the combinatorics of the free monoid.

## Contents



## Introduction

Infinite trees naturally arise in mathematical investigations on the semantics of programming languages. They arise in essentially two ways: when one *unloops* or *unfolds* a program undefinitely. One obtains then either a *tree of execution paths* (infinite in general) in the case of a program written in an imperative language like FORTRAN or an *expression tree* in the case of a program written in an applicative language like LISP. In the latter case, the expression tree is usually infinite although its value can be finitely computed in each case; this is possible by the use of **if–then–else** as a base function (like the addition of integers) and *not* as a piece of control structure. Once again, the infiniteness of the tree corresponds to the infiniteness of the set of possible computations.

In both cases, the semantics of the program is completely defined by the associated tree. Hence two programs are equivalent if the associated trees are the same (the converse being not true). Roughly speaking, this allows to distinguish between the equivalence of programs which is only due to the control structure (loops, recursive calls, etc. . . .) from the equivalence which also depends on the properties of the domains of computation and the given 'base' functions on these domains.

It should be noted that these infinite trees are finitely defined. Hence we are lead to try to decide whether two infinite trees defined in some finitary way are equal.

Two types of infinite trees will be considered: the *regular trees* which are defined by unlooping FORTRAN-like program or flowcharts and the *algebraic trees* which are defined by unfolding recursive program schemes more or less derived from LISP programs.

We shall introduce operations on trees: the *first-order substitution* which corresponds (roughly) to the sequential composition of flowcharts (by the operator; of ALGOL) or to functional application (in the case of an applicative language). We shall also introduce the *second-order substitution* which corresponds to the replacement of a function symbol in an expression tree by some expression tree intended to denote the corresponding function.

The theory of regular and algebraic trees will be developed for itself. The relevance to semantics will be shown with examples only in Sections 1.7 and 1.8.

Here is a brief survey of the content of the paper which is intended to be a synthesis of several aspects of infinite trees usually defined and studied separately for different purposes:

(1) *Topological* (i.e. metric) and *order-theoretical properties* of infinite trees are investigated in parallel in order to enlighten similarities and differences.

(2) *First-* and *second-order substitutions* are investigated in the two above frameworks.

(3) *Regular trees*, rational expressions defining them are studied. The concept of an iterative theory, due to C.C. Elgot, is one of the possible algebraic frameworks where to study infinite trees; the set of regular trees forms the free iterative theory. Regular trees also arise as most general first-order unifiers in a generalized sense.

(4) *Algebraic trees* play a similar role with respect to second-order substitutions as regular trees with respect to first-order ones. Their combinatorial properties are sufficiently complicated to yield an open problem equivalent to the DPDA equivalence problem.

The following text gives precise, unifying definitions of all the above topics. Many proofs are omitted either if they are too long and complex or if they are mere verifications from the definitions. Some of the given proofs are simpler than the original ones. Some of the stated results are 'new' in the sense that they have never been published before (to the author's knowledge) but are not really difficult to establish and were probably known from the specialists. (The present paper aims to be a reference for these technical lemmas and to facilitate future publications.) Other ones can be really claimed to be new: for instance Proposition 3.5.6 concerning second-order substitution and Theorem 5.9.1 saying that if the most general unifier of two algebraic trees exists, then it is algebraic.

This text has been written for the International Summer School on Theoretical Foundations of Programming Methodology, Marktoberdorf, West-Germany, August 1981, and a shortened version will appear under the same title in the proceedings to be published by D. Reidel Publishing Company, Dordrecht, Holland.

## 1. Basic definitions and examples

In this section we make precise some mathematical notations, we define finite and infinite trees over a ranked alphabet and we show informally how infinite trees can be associated with program schemes of various types.

### *1.1. Mathematical notations*

We denote by $\mathbf{N}$ the set of non-negative integers and by $\mathbf{N}_+$ the set of positive ones. We denote by $[n]$ the interval $\{1, 2, 3, \ldots, n\}$ for $n \geq 0$ (with $[0] = \emptyset$).

For sets $A$ and $B$ we denote by $A - B$ the set $\{a \in A \mid a \notin B\}$.

The *domain* of a partial mapping $f : A \to B$ will be denoted by $\mathbf{Dom}(f)$. The restriction of $f$ to a subset $A'$ of $A$ will be denoted by $f \restriction A'$.

If $f$ is a mapping $B^n \to C$ and $g_1, \ldots, g_n$ are mappings $A^m \to B$, we denote by $f \circ (g_1, \ldots, g_n)$ the mapping $h : A^m \to C$ such that $h(a_1, \ldots, a_m) = f(g_1(a_1, \ldots, a_m), \ldots, g_n(a_1, \ldots, a_m))$.

The set of total mappings: $A \to B$ will be denoted by $(A \to B)$ or by $B^A$.

### *1.2. Definitions*

In order to define trees, we shall use *ranked alphabets*. A ranked alphabet is a pair $(F, \rho)$ consisting of a set $F$, not necessarily finite, and a mapping $\rho : F \to \mathbf{N}$ which defines the *rank* of any symbol $f$ in $F$.

For such a set $F$, we denote by $F_i$ the set $\{f \in F \mid \rho(f) = i\}$, for $i \geq 0$.

In many cases the symbols in $F$ will be considered as *function symbols*; the rank of a function symbol is called its *arity* and a symbol of arity 0 is called a *constant symbol* (or a *constant*).

Following [41, 42, 62] we define a *tree over a ranked alphabet* $F$ (the rank function will always be $\rho$) as a partial mapping $t : \mathbf{N}_+^* \to F$ such that its domain is a *tree-domain* i.e. satisfies the following conditions:

**Dom**($t$) is *prefix-closed* (i.e. if $\alpha, \beta \in \mathbf{N}_+^*$, $\alpha\beta \in \mathbf{Dom}(t)$ then $\alpha \in \mathbf{Dom}(t)$) and not empty, (1.2.1)

if $\alpha \in \mathbf{N}_+^*$, $i, j \in \mathbf{N}$, $1 \leq i \leq j$ and $\alpha j \in \mathbf{Dom}(t)$ then $\alpha i \in \mathbf{Dom}(t)$. (1.2.2)

Furthermore we require the following condition which concerns the rank function:

if $t(\alpha) = f$ of arity $k \geq 0$ then, for $i \in \mathbf{N}_+$, $\alpha i \in \mathbf{Dom}(t)$ if and only if $1 \leq i \leq k$. (1.2.3)

We shall use the following terminology and notations:
- $M^\infty(F)$ for the set of all trees over $F$,
- $M(F)$ for the set of *finite* trees over $F$, i.e. of trees $t$ having a finite *set of nodes* **Dom**($t$),
- **First**($t$) for $t(\varepsilon)$, the label of the *root* of $t$,
- **Occ**($f, t$) for $\{\alpha \in \mathbf{Dom}(t) \mid t(\alpha) = f\}$, the set of *occurrences of f in t*,
- $t/\alpha$ for the *subtree of t issued from node* $\alpha$, i.e. the tree $t' = \lambda\beta \in \mathbf{N}_+^* . t(\alpha\beta)$,
- **Subtree**($t$) $= \{t/\alpha \mid \alpha \in \mathbf{Dom}(t)\}$,
- $|t| = \mathbf{Card}(\mathbf{Dom}(t))$ (an element of $\mathbf{N}_+ \cup \{\infty\}$) is the *size* of a tree $t$.

### *1.3. Examples*

Let $F = \{c, f, g, h, k, a, b, v_1, v_2\}$ with $\rho(c) = 3$, $\rho(f) = \rho(g) = 2$, $\rho(h) = \rho(k) = 1$, $\rho(a) = \rho(b) = \rho(v_1) = \rho(v_2) = 0$.

(1) Let $s$ be defined as follows:

$s(\varepsilon) = f$,

$s(1) = s(11) = k$, $s(111) = a$,

$s(2) = h$, $s(21) = b$.

It is depicted in Fig. 1.

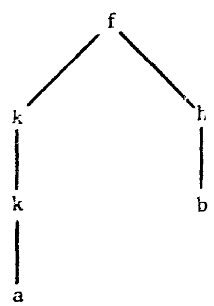


Fig. 1.

(2) Let now $t$ be defined by

$t(\varepsilon) = f$, $t(1) = a$,

$t(2^n) = g$, $t(2^n 1) = b$ for all $n \geq 1$.

This infinite tree is shown in Fig. 2. The subtree $t/2$ consists of all the $g$'s and $b$'s of $t$.

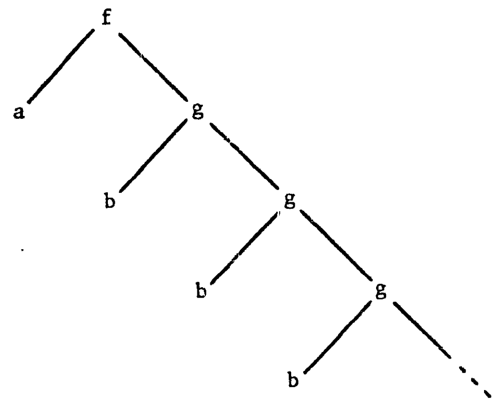


Fig. 2.

(3) The tree $u$ such that $u(1^n) = h$ for all $n \geq 0$ consists of one infinite branch. We identify it with the infinite word $h^\omega$. See Nivat [54] and section 5.10.

(4) Our last example will be the tree $w$ of Fig. 3. It can be defined as follows:

$$w(3^n) = c, \qquad w(3^n 1) = v_1,$$

$$w(3^n 21^n) = v_2, \qquad w(3^{n+1} 21^m) = h \quad \text{for all } n \geq m \geq 0.$$

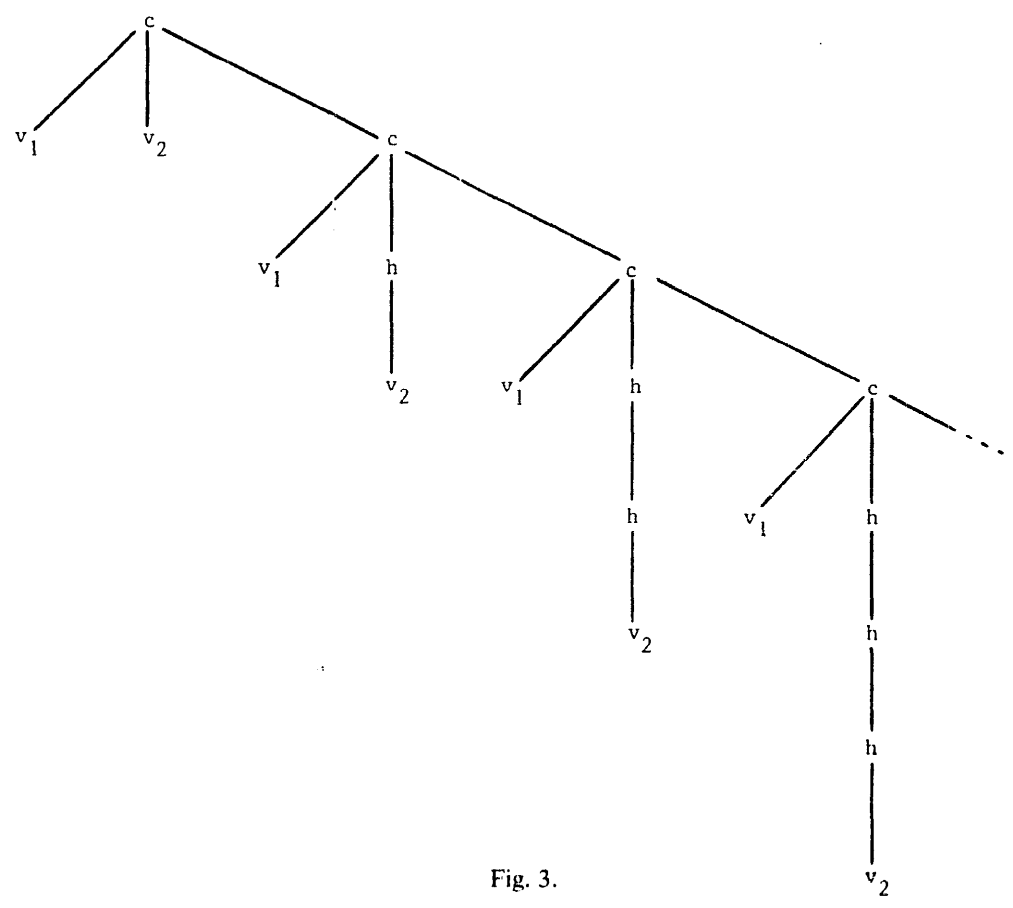


Fig. 3.

If we consider $F$ as a set of function symbols, the finite trees over $F$ can be identified with *well-formed terms over F* and written linearly with commas and parentheses. For instance, the tree $s$ of Fig. 1 can be written

$$f(k(k(a)), h(b))$$

From such a notation, one can infer the arities of the symbols $f$, $k$, $a$, $h$, $b$.

We shall also omit the parentheses surrounding the arguments of *monadic* (i.e. of arity 1) function symbols; the above tree can also be written

$$f(kka, hb) \quad \text{or} \quad f(k^2a, hb).$$

Within a proof or a theorem, we shall only write down *well-formed* trees and terms; hence when declaring "let $t$ be of the form $f(t_1, t_2, \ldots, t_n) \ldots$" we also declare that $f$ is of arity $n$. And this allows $n$ to be 0 (in this case $(t_1, \ldots, t_n)$ is the empty sequence, i.e. $t = f$).

### 1.4. F-magmas, F-algebras

The *standard operation* defined by a symbol $f$ of $F_k$, $k \geq 0$ is the mapping $\bar{f}: M^\infty(F)^k \to M^\infty(F)$ such that $\bar{f}(t_1, \ldots, t_k) = t'$ where

$$t'(\varepsilon) = f,$$

$$t'(i\alpha) = t_i(\alpha) \quad \text{if } 1 \leq i \leq k,$$

$t'(\alpha)$ is undefined otherwise.

The mapping $\bar{f}$ maps $M(F)^k$ into $M(F)$. The notation of finite trees with commas and parentheses allows us to write

$$\bar{f}(t_1, \ldots, t_k) = f(t_1, \ldots, t_k),$$

i.e. to identify $\bar{f}$ with $f$.

We shall do the same for infinite trees and specify finite or infinite trees by "let $t = f(t_1, \ldots, t_k)$ for $(t_1, \ldots, t_k)$ in $M^\infty(F) \ldots$".

The definition of the $\bar{f}$'s makes $M(F)$ and $M^\infty(F)$ into *F-magmas* (equivalently *F-algebras* [41] but we prefer the former terminology since the terms 'algebra', 'algebraic' are overused in mathematics). We shall *not* distinguish between the sets of trees $M(F)$ (or $M^\infty(F)$) and the associated magmas, and we shall frequently do the same for arbitrary $F$-magmas.

We shall talk of *F-homomorphism* to specify that the magma structure under consideration is relative to $F$ when this is not completely clear from the context.

It is well known that $M(F)$ is an initial $F$-magma hence *the* initial $F$-magma. We shall denote by $\mathbf{eval}_A$ the unique morphism of $M(F)$ into an $F$-magma $A$ (we consider $t$ as a syntactic object denoting the element $\mathbf{eval}_A(t)$ of $A$).

We shall refer to the $\bar{f}$'s as the *F-operations* on $M(F)$ (and on $M^\infty(F)$) to contrast them with other operations to be introduced below.

### *1.5. Basic lemmas*

Let us point out some basic facts upon which most proofs and definitions concerning trees are based.

**Lemma 1.5.1.** *Let $t$ and $t'$ be trees in $M^\infty(F)$ of the forms $t = f(t_1, \ldots, t_k)$, $t' = g(t'_1, \ldots, t'_l)$. They are equal if and only if $f = g$ (whence $k = l$) and $t_i = t'_i$ for all $i = 1, \ldots, k$.*

The next lemmas hold for $M(F)$ only and follow from its initiality property; they will be extended to $M^\infty(F)$ in the next section by means of topological and order-theoretical considerations.

**Lemma 1.5.2** (Proof by structural induction). *In order to prove a property of the form $\forall t \in M(F) \,.\, P(t)$, it suffices to prove:*

(1) $\forall t \in F_0 \,.\, P(t)$,

(2) $\forall k > 0, \forall f \in F_k, \forall t_1, \ldots, t_k \in M(F) \,.\, P(t_1)$ **and** $\ldots$ **and** $P(t_k) \Rightarrow P(f(t_1, \ldots, t_k))$.

**Lemma 1.5.3** (Definition by structural induction). *There exists one and only one mapping $\varphi : M(F) \to A$ such that*

(1) *$\varphi(f) = a(f)$ for all $f \in F_0$,*

(2) *$\varphi(f(t_1, \ldots, t_k)) = b(f, \varphi(t_1), \ldots, \varphi(t_k))$ for all $k > 0$, all $f \in F_k$, all $t_1, \ldots, t_k$ in $M(F)$*

*where $a$ and $b$ are given mappings: $F_0 \to A$ and $F_1 \times M(F)^1 \cup \cdots \cup F_k \times M(F)^k \cup \cdots \to A$ respectively.*

**Proofs.** Immediate application of the initiality property to the $F$-magmas $\{u \in M(F) \mid P(u)$ is true$\}$ equipped with the standard operations for Lemma 1.5.2 and $A$ equipped with $F$-operations obviously constructed from $a$ and $b$ for Lemma 1.5.3. □

**Notations 1.5.4.** For $T \subseteq M^\infty(F)$ and $G \subseteq F$ we denote by $G(T)$ the set of trees of the form $f(t_1, \ldots, t_n)$ for $f \in G$ and $t_1, \ldots, t_n \in T$ and by $M(G, T)$ the least sub-$G$-magma of $M^\infty(F)$ which contains $T$.

### *1.6. Representations of trees by languages*

The definitions given in Section 1.2 show that a tree $t$ can be *represented* i.e. completely defined by an indexed family of languages $(\mathbf{Occ}(f, t))_{f \in F}$ which reduces to a tuple if the number of symbols of $F$ occurring in $t$ is finite, or by the single language $L(t) = \{\alpha f \mid \alpha \in \mathbf{Occ}(f, t), f \in F\} \subseteq \mathbf{N}_+^* F$.

Another representation has been defined by Rosen [63], and investigated in depth by Courcelle [15]. One represents a tree $t$ by the language $\mathbf{Brch}(t)$ of its *branches* which is defined as follows.

A new alphabet $\bar{F}$ is associated with $F$ by

$$\bar{F} = \{[f, i] \mid f \in F,\ 1 \leq i \leq \rho(f)\} \cup F_0.$$

Let $t$ be a tree in $M^\infty(F)$.

For every $a \in F_0$, every $\alpha \in \mathbf{Occ}(a, t)$, let $\bar{\alpha}$ be the word

$$\bar{\alpha} = [f_1, i_1][f_2, i_2] \dots [f_n, i_n]a$$

where

$$\alpha = i_1 i_2 \dots i_n, \qquad f_j = t(i_1 i_2 \dots i_{j-1}) \quad \text{for } 1 \leq j \leq n.$$

Then we define $\mathbf{Brch}(t)$ as the set of all such words $\bar{\alpha}$.

Note in particular that $\mathbf{Brch}(t) = \emptyset$ if $t$ has no occurrence of any symbol in $F_0$. Hence $\mathbf{Brch}(t)$ does not represent *all* infinite trees, but only the *locally finite* ones (in the sense of [15]). The set

$$M^{\mathrm{loc}}(F) = \{t \in M^\alpha(F) \mid \text{for all } \alpha \text{ in } \mathbf{Dom}(t), \text{ there exists } \beta \text{ such that } t(\alpha\beta) \in F_0\}$$

is the set of locally finite trees.

Note that $M(F) \subsetneq M^{\mathrm{loc}}(F)$.

In certain circumstances, it is useful to be able to describe the *infinite branches* of trees.

If $t \in M^\infty(F)$ and $\alpha$ is an infinite word in $\mathbb{N}_+^\omega$ every finite prefix of which is in $\mathbf{Dom}(t)$, then we associate with $\alpha$ the infinite word $\bar{\alpha}$ of $\bar{F}^\omega$ such that

$$\bar{\alpha} = [f_1, i_1][f_2, i_2][f_3, i_3] \dots [f_n, i_n] \dots,$$

$$\alpha = i_1 i_2 i_3 \dots i_n \dots,$$

$$f_j = t(i_1 i_2 \dots i_{j-1}) \quad \text{for all } j \geq 1.$$

We let $\mathbf{Brch}^\omega(t)$ denote the set of all such infinite words $\bar{\alpha}$ and we let

$$\mathbf{Brch}^\infty(t) = \mathbf{Brch}(t) \cup \mathbf{Brch}^\omega(t).$$

Proposition 1.6.2 below shows that $\mathbf{Brch}^\infty(t)$ uniquely defines $t$ in all cases.

Those who do not like infinite words can use instead of $\mathbf{Brch}^\infty(t)$ the language $\mathbf{PBrch}(t)$ consisting of all the finite prefixes of all words in $\mathbf{Brch}^\infty(t)$ (so that $\mathbf{Brch}(t) \subset \mathbf{PBrch}(t)$).

**Example 1.6.1.** Let us use the trees of Example 1.3.

(1) $L(s) = \{f, 1k, 11k, 111a, 2h, 21b\}$,

$$\mathbf{Brch}(s) = \{f_1kka, f_2hb\}$$

(we use $f_i$ for $[f, i]$ when $\rho(f) \geq 2$ and $f$ for $[f, 1]$ when $\rho(f) = 1$ in our examples for more clarity).

$$\mathbf{Brch}^\omega(s) = \emptyset.$$

(2) $\mathbf{Brch}(t) = \{f_1 a, f_2 g_2^n g_1 b \mid n \geq 0\}$.

Note that $t$ is locally finite. Its language of branches is regular. We shall see that $t$ is a regular tree.

(3) $L(u) = \{1^n h \mid n \geq 0\}$,

$\mathbf{Brch}(u) = \emptyset$,

$\mathbf{Brch}^\infty(u) = \mathbf{Brch}^\omega(u) = \{hhh \ldots\} = \{h^\omega\}$.

This tree is not locally finite. It is also a regular tree.

(4) $\mathbf{Brch}(w) = \{c_3^n c_1 v_1, c_3^n c_2 h^n v_2 \mid n \geq 0\}$.

This tree is locally finite. Its language of branches is context-free. We shall see that $w$ is an algebraic tree.

**Proposition 1.6.2.** *For $t$, $t'$ in $M^\infty(F)$:*
(1) *$t = t'$ if and only if $L(t) = L(t')$,*
(2) *$t = t'$ if and only if* $\mathbf{Brch}^\infty(t) = \mathbf{Brch}^\infty(t')$,
(3) *$t = t'$ if and only if* $\mathbf{PBrch}(t) = \mathbf{PBrch}(t')$,
(4) *if $t \in M^{\mathrm{loc}}(F)$, then $t = t'$ if and only if* $\mathbf{Brch}(t) = \mathbf{Brch}(t')$.

It follows that each family of languages $\mathscr{C}$ naturally defines three families of trees:

$$L(\mathscr{C}) = \{t \in M^\infty(F) \mid L(t) \in \mathscr{C}\},$$

$$P(\mathscr{C}) = \{t \in M^\infty(F) \mid \mathbf{PBrch}(t) \in \mathscr{C}\},$$

$$B(\mathscr{C}) = \{t \in M^{\mathrm{loc}}(F) \mid \mathbf{Brch}(t) \in \mathscr{C}\}.$$

Certain classes of trees can be characterized in this way (see Sections 4.11 and 5.5 and Damm [30]).

### 1.7. *Flowchart schemes and infinite trees*

Consider the flowchart scheme $S$ of Fig. 4. Its infinite unlooping yields the tree of Fig. 2.

An *interpretation* $\mathbf{I}$ *for* $S$ is an object $\mathbf{I} = \langle D_\mathbf{I}, f_\mathbf{I}, g_\mathbf{I}, a_\mathbf{I}, b_\mathbf{I}\rangle$ consisting of
- a nonempty set $D_\mathbf{I}$,
- partial mappings $a_\mathbf{I}, b_\mathbf{I} : D_\mathbf{I} \rightharpoonup D_\mathbf{I}$,
- partial mappings $f_\mathbf{I}, g_\mathbf{I} : D_\mathbf{I} \rightharpoonup D_\mathbf{I} \times \{1, 2\}$ (the 1 and 2 correspond respectively to the left and right exits of *actions* $f$ and $g$).

There corresponds to $S$ and $\mathbf{I}$ a partial mapping $S_\mathbf{I} : D_\mathbf{I} \to D_\mathbf{I}$ defined by $S_\mathbf{I}(d) = d'$ if and only if there exists a sequence $d_0, d_1, d_2, \ldots, d_n$ of elements of $D_\mathbf{I}$ with $d_0 = d$, $d_n = d'$, which corresponds to a *computation* of $S$ in $\mathbf{I}$.

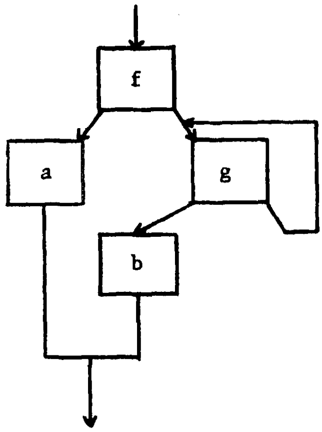


Fig. 4.

We do not formally define a computation but we give a typical example for the above scheme $S$:

$$(d_0, d_1, d_2, d_3, d_4, d_5)$$

where

$$f_{\mathbf{I}}(d_0) = (d_1, 2), \qquad g_{\mathbf{I}}(d_1) = (d_2, 2),$$

$$g_{\mathbf{I}}(d_2) = (d_3, 2), \qquad g_{\mathbf{I}}(d_3) = (d_4, 1), \qquad b_{\mathbf{I}}(d_4) = d_5.$$

Note that $f_{\mathbf{I}}$ and $g_{\mathbf{I}}$ are not only 'tests' since they can modify their data (see the interpretation **I** defined below).

Let $P$ be the following program (with integer variables):

```
begin x ← x + 3;
      y ← 2x + 7;
      if x ≤ y
      then y ← 27x + 3;
      else begin do x ← x − 8;
                    y ← 10x;
                 until x ≤ 0
                 od;
                 x ← 82y;
                 y ← 0;
           end
      fi
end
```

We do not claim that $P$ computes anything interesting, but we chose it only as an example.

It 'corresponds' to the pair $(S, \mathbf{I})$ consisting of flowchart scheme $S$ and interpretation $\mathbf{I}$ defined as follows:

$D_{\mathbf{I}} = \mathbf{Z}^2$,

$a_{\mathbf{I}}((x, y)) = (x', y')$ with $y' = 27x + 3$ and $x' = x$,

$b_{\mathbf{I}}((x, y)) = (x', y')$ with $x' = 82y$ and $y' = 0$,

$f_{\mathbf{I}}((x, y)) = ((x', y'), i)$ if $x' = x + 3$, $y' = 2x' + 7$ and $i = 1$ if $x' \leq y'$, $i = 2$ if $y' < x'$,
equivalently, if $x' = x + 3$, $y' = 2x + 13$ and $i = 1$ if $x \geq -10$, $i = 2$ if $x < -10$

$g_{\mathbf{I}}((x, y)) = ((x', y'), i)$ with $x' = x - 8$, $y' = 10x'$ and $i = 1$ if $x' \leq 0$, $i = 2$ if $x' > 0$,
equivalently if $x' = x - 8$, $y' = 10x - 80$ and $i = 1$ if $x \leq 8$, $i = 2$ if $x > 8$.

By 'corresponds' we mean in particular that the function computed by $P$ is $S_{\mathbf{I}}$.

The tree of Fig. 2, let us call it $t(S)$, can also be seen as an 'infinite' flowchart scheme from which a partial function $t(S)_{\mathbf{I}}$ can be defined as for finite schemes.

The two main facts are the following ones:

(1) $S_{\mathbf{I}} = t(S)_{\mathbf{I}}$ for all interpretations $\mathbf{I}$,

(2) for any two schemes $S$ and $S'$: $S_{\mathbf{I}} = S'_{\mathbf{I}}$ for all interpretations $\mathbf{I}$, i.e. $S$ and $S'$ are *equivalent*, if and only if $t(S) = t(S')$.

This model of computation has been introduced by Elgot [32, 33]. Infinite trees have been used by Cousineau [28, 29], Casteran [13] and Enjalbert [36] in order to study programs, their proofs and transformations in terms of program schemes.

### *1.8. Recursive program schemes and infinite trees*

Let us consider the following (fancy) recursive definition:

$$K(x_1, x_2) = \textbf{if } x_1 = x_2 \textbf{ then } x_1 + 3 \textbf{ else } 18 \cdot K(x_1, x_2 - 1).$$

It can be considered as an instance of the following recursive program scheme:

$$\varphi(v_1, v_2) = c(v_1, v_2, \varphi(v_1, h(v_2)))$$

for the *interpretation* $\mathbf{I} = \langle D_{\mathbf{I}}, c_{\mathbf{I}}, h_{\mathbf{I}} \rangle$ consisting of a *domain* $D_{\mathbf{I}} = \mathbf{Z} \cup \{\perp\}$ ($\perp$ means 'undefined') and functions

$$c_{\mathbf{I}} = \lambda x, y, z \in D_{\mathbf{I}}[\textbf{if } x = y \textbf{ then } x + 3 \textbf{ else } 18z],$$

$$h_{\mathbf{I}} = \lambda x \in D_{\mathbf{I}}.[x - 1]$$

which give meaning to the function symbols $c$ and $h$.

A formal computation of $\varphi$ i.e. a infinite unfolding of the recursion leaving $c$ and $h$ unevaluated yields the infinite tree of Fig. 3, let us denote it by $t(\varphi(v_1, v_2))$.

We can consider it as an infinite well-formed expression denoting the function computed by $\varphi$ in every interpretation. As in the case of flowcharts:

(1) $K = \varphi_{\mathbf{I}}$, the function computed by $\varphi$ in $\mathbf{I}$ and this function can be defined from $t(\varphi(v_1, v_2))$,

(2) for any two recursive program schemes $\varphi(v_1, v_2)$ and $\varphi'(v_1, v_2)$, the functions $\varphi_{\mathbf{I}}$ and $\varphi'_{\mathbf{I}}$ are the same for all interpretations $\mathbf{I}$, i.e. $\varphi$ and $\varphi'$ are *equivalent*, if and only if $t(\varphi(v_1, v_2)) = t(\varphi'(v_1, v_2))$.

These facts are investigated in depth in many works by Courcelle, Nivat, Guessarian [24, 25, 44, 53).

It is useful to relativize the equivalence of program schemes to classes of interpretations in order to get closer to the equivalence of programs. Classes of interpretations for these kinds of program schemes are investigated in [11, 18, 22, 24, 25, 39, 44].

## 2. Topological and order-theoretical properties of trees

We show that the set of infinite trees can be considered as a compact metric space and also as a complete partial order. In both cases an infinite tree can be considered as the limit (in some sense) of a sequence of finite trees and this allows to extend 'continuous' mappings from finite trees to infinite ones.

Hence a double theory of infinite trees can be developed either in the framework of topology or in that of the theory of ordered sets. In particular two universal characterizations of the $F$-magma of infinite trees can be given.

Since the introduction of infinite trees has been motivated by studies in semantics of programming languages (via program schemes) the order-theoretical approach has been developed first. It seems better suited for *semantics* (but this was not the opinion of Elgot since his theory of *monadic computations* [32] avoids orderings; Arnold and Nivat also avoid orderings in [2]).

In the present paper where we investigate *syntactical* properties of trees, both of them are useful.

### 2.1. *Contracting magmas*

Let $(E, d)$ be a metric space. For $f: E \to E$ let us denote by $\|f\|$ the least upper bound of $\{d(f(x), f(x'))/d(x, x') \mid x, x' \in E, x \neq x'\}$. A mapping $f: E \to E$ is *contracting* if there exists a real number $c$, $0 \leq c < 1$ such that, for all $x, x'$ in $E$,

$$d(f(x), f(x')) \leq c \cdot d(x, x'),$$

i.e. if $\|f\| < 1$.

A contracting mapping is uniformly continuous. For $k \geq 2$ we shall also denote by $d$ the distance on $E^k$ which is defined by

$$d((t_1, \ldots, t_k), (t'_1, \ldots, t'_k)) = \mathbf{Max}\{d(t_i, t'_i) \mid 1 \leq i \leq k\}.$$

An $F$-magma $A = \langle A, (f_A)_{f\in F}\rangle$ is *contracting* if

(1) $A$ is a complete metric space with distance $d_A$,

(2) $d_A(x, y) \leq 1$ for all $x, y \in A$,

(3) $f_A$ is contracting for all $f$ in $F$ and

(4) $\|A\| = \text{Sup}\{\|f_A\| \mid f \in F\} < 1$.

A *morphism* of contracting $F$-magmas is a morphism of $F$-magmas which is uniformly continuous.

We recall a well-known theorem which justifies our interest in contracting mappings:

**Fix-point Theorem 2.1.1.** *Let $E$ be a complete metric space. Every contracting mapping $f: E \to E$ has a unique fix-point.*

**Proof.** We are to show the existence and unicity of $x$ in $E$ such that $f(x) = x$.

Assume we have two such fix-points $x$ and $x'$. Then $d(x, x') = d(f(x), f(x'))$ since $x$ and $x'$ are fix-points and $d(f(x), f(x')) \leq c \,.\, d(x, x')$ since $f$ is contracting. Hence $d(x, x') = 0$ and $x = x'$. For the existence, let $x_0$ be any element of $E$. Let $x_n = f^n(x_0)$. We have

$$d(x_{n+1}, x_n) \leq c^n \cdot d(x_1, x_0),$$

$$d(x_{m+n}, x_n) \leq c^n(c^{m-1} + c^{m-2} + \cdots + 1)d(x_1, x_0)$$

$$\leq c^n(1-c)^{-1} \cdot d(x_1, x_0)$$

hence $(x_n)_{n\geq 0}$ is a Cauchy sequence. It has a limit $x$ and $x = f(x)$, by continuity. □

*2.2. $M^\infty(F)$ as a metric space*

Let $t$ and $t'$ be two elements of $M^\infty(F)$. Let us define

$$\delta(t, t') = \begin{cases} \infty & \text{if } t = t', \\ \text{Min}\{|\alpha| \mid \alpha \in \text{Dom}(t) \cap \text{Dom}(t'), t(\alpha) \neq t'(\alpha)\} & \text{if } t \neq t'. \end{cases}$$

Finally we let

$$d(t, t') = \begin{cases} 0 & \text{if } t = t', \\ 2^{-\delta(t,t')} & \text{if } t \neq t'. \end{cases}$$

It is easy to show that $d$ is a *distance* on $M^\infty(F)$ making it into a complete metric space [3, 9, 5]. This distance is even *ultrametric*. It is essentially the same as the distance that one puts on the ring of formal power series.

Note that $d(t, t') \leq 1$ for all $t, t'$ in $M^\infty(F)$.

It can be shown that $M^\infty(F)$ is compact if and only if $F$ is finite, that $M(F)$, the set of finite trees is a *dense* subset of $M^\infty(F)$ and that $M^\infty(F)$ is the topological completion of $M(F)$ (Mycielski and Taylor [52], Arnold and Nivat [3]).

**Proposition 2.2.1.** *Every uniformly continuous mapping: $M(F)^k \to E$ where $E$ is complete extends uniquely into a uniformly continuous mapping: $M^\infty(F)^k \to E$.*

**Remark.** This also applies to a property $P(t)$ for $t$ in $M^\infty(F)$ such that

(1) $\forall t \in M(F) . P(t)$

(see Lemma 1.5.2) and which is *continuous* in the sense that

(2) if $t = \mathbf{Lim}_{n\to\infty} t_n$ where $t, t_0, \ldots, t_n, \ldots, \in M^\infty(F)$
and $\forall n \in \mathbf{N} . P(t_n)$ holds then $P(t)$ holds too.

(i.e. which defines a continuous mapping from $M^\infty(F)$ into the discrete space {**true**, **false**}).

From (1) and (2) one can conclude that $P(t)$ holds for all $t$ in $M^\infty(F)$.

Let us now consider the $F$-operations on $M^\infty(F)$.

The mappings $\bar{f}: M^\infty(F)^k \to M^\infty(F)$ are contracting (with $\|f\| = \frac{1}{2}$ if $\rho(f) \geq 1$, $\|f\| = 0$ if $\rho(f) = 0$) hence

**Proposition 2.2.2.** *$M^\infty(F)$ is a contracting $F$-magma.*

Let us answer to the natural question:

**Proposition 2.2.3.** *$M^\infty(F)$ is the initial contracting $F$-magma.*

**Proof.** See Bloom and Patterson [9] where a very similar result is proved. □

**Remark 2.2.4.** The hypothesis that $\mathbf{Sup}\{\|f_A\| \mid f \in F\} < 1$ that we made in the definition of a contracting $F$-magma is essential to insure Proposition 2.2.3.

This hypothesis is not made in [9]. It follows that $M^\infty(F)$ is initial (in the corresponding category) if and only if $F - F_0$ is finite and that there is no initial object if $F - F_0$ is infinite.

Proposition 2.2.3 says that a tree $t$ in $M^\infty(F)$ can be seen as a syntactic object denoting an element of $A$, where $A$ is a contracting $F$-magma. We denote it by $\mathbf{eval}_A(t)$. Hence we also denote by $\mathbf{eval}_A$ the unique uniformly continuous extension to $M^\infty(F)$ of the mapping $\mathbf{eval}_A : M(F) \to A$ defined in Section 1.4).

## *2.3. Complete magmas*

An *ω-complete $F$-magma $A$* is an $F$-magma equipped with a partial order $\leq_A$ such that

(1) $A$ has a least element,

(2) every countable directed subset $B$ (equivalently every increasing sequence) has a least upper bound $\mathbf{Sup}(B)$, and

(3) the functions $f_A$'s are monotone and $\omega$-continuous (i.e. preserve the **Sup**'s of countable directed subsets).

Hence this concept coincides with that of *$\omega$-continuous F-algebra* introduced in [41].

We call *A complete* if the least upper bounds are taken with respect to arbitrary directed sets in (2) and (3).

All properties we shall state below hold for both completeness concepts. The $\omega$-completeness will be sufficient for dealing with trees.

We refer the reader to [26, 70] for more details about partial orders and other possible concepts of completeness.

It is certainly not necessary to give the proof of the following well-known theorem:

**Tarski fix-point theorem 2.3.1.** *Let E be an $\omega$-complete partial order with least element e; let $f: E \to E$ be $\omega$-continuous. The element $u_0 = \mathbf{Sup}\{f^n(e) \mid n \geq 0\}$ of E is the least fix-point of f in E and also, the least solution in E of the inequation $f(u) \leq u$.*

We shall denote $u_0$ by $\mu x . f(x)$.

This applies to systems of equations since a system $S = \langle x_i = f_i(x_1, \ldots, x_n);\ 1 \leq i \leq n\rangle$ where $x_i \in E_i$ for $1 \leq i \leq n$ can be considered as a single equation $x = f(x)$ to be solved in $E_1 \times E_2 \times \cdots \times E_n$ with $f((d_1, \ldots, d_n)) = (f_1(d_1, \ldots, d_n), \ldots, f_n(d_1, \ldots, d_n))$ for all $d_1$ in $E_1, \ldots, d_n$ in $E_n$.

Another fundamental lemma is the following one.

**Lemma 2.3.2** (Bekič [4], Leszcylowski [50]). *Let E and E' be two $\omega$-complete partial orders, let $f: E \times E' \to E$ and $g: E \times E' \to E'$ be $\omega$-continuous.*

(1) *The mapping $h: E' \to E$ defined by $h(y) = \mu x . f(x, y)$ is $\omega$-continuous.*

(2) *The two systems $S = \langle x = f(x, y),\ y = g(x, y)\rangle$ and $S' = \langle x = h(y), y = g(h(y), y)\rangle$ have the same least solution in $E \times E'$.*

In other words the least solution $(x_0, y_0)$ of $S$ can be defined by $y_0 = \mu y . g(h(y), y)$ and $x_0 = h(y_0) = \mu x . f(x, y_0)$.

**Proof.** Part (1) follows from the fact that $h$ is the least upper bound of the sequence of $\omega$-continuous functions $h_n$, $n \geq 0$ such that

$$h_n(y) = f(f(f(\ldots, f(e, y), \ldots, y)y))$$

(with $n$ occurrences of $f$).

Let us sketch the proof of part (2).

Let $y_1 = \mu y . g(h(y), y)$. It is easy to verify that $(h(y_1), y_1)$ is a solution of $S$, hence $(x_0, y_0) \leq (h(y_1), y_1)$.

Since $(x_0, y_0)$ is a solution of $S$, $x_0 = f(x_0, y_0)$ hence $h(y_0) \leq x_0$ by definition of $h$. Hence $g(h(y_0), y_0) \leq g(x_0, y_0) = y_0$. Hence $y_1 = \mu y . g(h(y), y) \leq y_0$. Since $h$ is monotone, $h(y_1) \leq h(y_0) \leq x_0$. Hence $(h(y_1), y_1) \leq (x_0, y_0)$.

Hence we have shown that $(x_0, y_0) = (h(y_1), y_1)$. □

### 2.4. $M_\Omega^\infty(F)$ as an $\omega$-complete partial order

Let $F$ be a ranked alphabet. Let $\Omega$ be a new symbol of arity 0 that we add to $F$.

For any complete $F$-magma $A$ we shall define the value $\Omega_A$ of $\Omega$ as the least element of $A$.

Since $\Omega$ will play a special role, we shall use the notations

$$M_\Omega(F) \text{ for } M(F \cup \{\Omega\}) \quad \text{and} \quad M_\Omega^\infty(F) \text{ for } M^\infty(F \cup \{\Omega\}).$$

We define a partial order on $M_\Omega^\infty(F)$ denoted by $\leqslant$ as follows:

$t \leqslant t'$ if and only if $\mathbf{Dom}(t) \subseteq \mathbf{Dom}(t')$ and for all $\alpha$ in $\mathbf{Dom}(t)$, if $t(\alpha) \neq \Omega$ then $t'(\alpha) = t(\alpha)$.

It is fairly easy to show that $\leqslant$ is a partial order, that $\Omega$ is the least element of $M_\Omega^\infty(F)$ with respect to $\leqslant$.

Every directed subset $A$ of $M_\Omega^\infty(F)$ has a least upper bound, $a = \mathbf{Sup}(A)$ in $M_\Omega^\infty(F)$ defined by

$$\mathbf{Dom}(a) = \bigcup \{\mathbf{Dom}(t) \mid t \in A\},$$

and for all $\alpha$ in $\mathbf{Dom}(a)$,

$$a(\alpha) = \begin{cases} f \in F & \text{if } t(\alpha) = f \text{ for some } t \text{ in } A, \\ \Omega & \text{if } t(\alpha) = \Omega \text{ for all } t \text{ in } A \text{ such that } \alpha \in \mathbf{Dom}(t). \end{cases}$$

The mappings $\bar{f}$ are monotone and $\omega$-continuous hence we can conclude that $M_\Omega^\infty(F)$ is an $\omega$-complete $F$-magma (in fact a complete $F$-magma as well).

The following proposition is analogous to Proposition 2.2.3:

**Proposition 2.4.1** ([41]). *$M_\Omega^\infty(F)$ is the initial $\omega$-complete $F$-magma.*

**Proposition 2.4.2.** *Let $E$ be an $\omega$-complete partial order. Every monotone mapping $h : M_\Omega(F)^k \to E$ can be uniquely extended into an $\omega$-continuous mapping: $M_\Omega^\infty(F)^k \to E$.*

Hence, if $A$ is an $\omega$-complete $F$-magma the monotone mapping $\mathbf{eval}_A : M_\Omega(F) \to A$ extends uniquely to $M_\Omega^\infty(F)$. We also denote by $\mathbf{eval}_A$ its extension. This means that a tree $t$ in $M_\Omega^\infty(F)$ denotes an element $(\mathbf{eval}_A(t))$ of $A$.

**Remark 2.4.3.** Let us call a property $P(t)$ of trees in $M_\Omega^\infty(F)$ *$\omega$-continuous* if $P(t)$ is true whenever it is true for all element of an increasing sequence $t_n$ in $M_\Omega^\infty(F)$ with least upper bound $t$ (i.e. if $P$ is monotone and $\omega$-continuous as a mapping: $M_\Omega^\infty(F) \to \{\mathbf{true}, \mathbf{false}\}$ with $\mathbf{false} < \mathbf{true}$).

If $P$ is $\omega$-continuous, in order to establish $\forall t \in M_\Omega^\infty(F)\, .\, P(t)$, it suffices to establish the validity of

$$\forall t \in M_\Omega(F)\, .\, P(t)$$

for instance by structural induction (Lemma 1.5.2).

Let us close this section with a convention: the words 'continuous' and 'complete' will refer to the topological approach whereas '$\omega$-continuous' and '$\omega$-complete' will refer to the order-theoretical one.

## 3. Substitutions

By introducing *variables* we shall make trees denote *functions* and not only *values* as we did up to now.

Then we shall define the *first-order substitution*, i.e. the substitution of trees for variables in other trees as a syntactic counterpart of the composition of functions and we shall state its basic properties.

We shall also introduce the *second-order substitution*, i.e. the substitution of trees for function symbols in trees. This corresponds to replacing a function name by its definition everywhere it occurs in some tree.

When reducing trees to words (if $\rho(f)=1$ for all $f$ in $F$) the first-order substitution reduces to the concatenation of words whereas the second-order one reduces to the homomorphism.

We shall use (possibly infinite) ranked alphabets $F$ and $G$, not necessarily disjoint or distinct.

### *3.1. Trees with variables*

Let $V$ be a set of *variables* i.e. of symbols of arity 0. By using them together with $F$ we can define the following sets of trees:

$M(F \cup V)$ also denoted by $M(F, V)$,

$M^{\infty}(F \cup V)$ also denoted by $M^{\infty}(F, V)$

and similarly for $M_{\Omega}(F, V)$ and $M^{\infty}_{\Omega}(F, V)$.

When using the notation $M^{\infty}(F \cup V)$ we use the elements of $V$ as constants, whereas we emphasize their special role (see below) when we use the notation $M^{\infty}(F, V)$.

If we need an enumeration of $V$ we shall take $V=\{v_1, v_2, v_3, \ldots, v_n, \ldots\}$, $V_k=\{v_1, \ldots, v_k\}$ and $V_0=\emptyset$.

Alternative sets of variables will be $W, X, Y$.

For $t$ in $M^{\infty}(F, V)$ we define $\mathbf{Var}(t)$, the set of variables from $V$ occurring in $t$, i.e. $\mathbf{Var}(t)=\{v \in V \mid \mathbf{Occ}(v, t) \neq \emptyset\}$.

### *3.2. Derived operators*

Let $A$ be an $F$-magma.

It is clear that a tree $t$ in $M(F, V_k)$ can be seen as denoting a mapping: $A^k \to A$. Such a mapping is called a *derived operator* (derived from the $F$-operators) and is denoted by $\mathbf{derop}_A(t)$.

The mapping $\mathbf{derop}_A(t)$ can be defined 'point-wise' as follows:

$$\mathbf{derop}_A(t)(a_1, \ldots, a_k) = \mathbf{eval}_{A'}(t),$$
$$A' = \langle A, (f_A)_{f \in F}, (\bar{v})_{v \in V_k} \rangle, \quad (3.2.1)$$
$$\bar{v}_i = a_i \quad \text{for } i = 1, \ldots, k,$$

or 'globally' by structural induction:

$$\mathbf{derop}_A(v_i) \text{ is the } i\text{th projection: } A^k \to A,$$
$$\mathbf{derop}_A(f) \text{ is the constant function equal to } f_A \text{ for } f \in F_0, \quad (3.2.2)$$
$$\mathbf{derop}_A(f(t_1, \ldots, t_n)) = f_A \circ (\mathbf{derop}_A(t_1), \ldots, \mathbf{derop}_A(t_n)).$$

Definition (3.2.1) says that $M(F, V_k)$ is the *free F-magma* generated by $V_k$. It could easily be extended to the case of an infinite set of variables $V$ instead of $V_k$, showing the existence of a mapping

$$\mathbf{derop}_A(t): A^V \to A$$

for all $t$ in $M(F, V)$.

It could also be extended to the case of an $\omega$-complete $F$-magma $A$ and $t$ in $M_\Omega^\infty(F, V)$ since the corresponding $F$-magma $A'$ is also $\omega$-complete and $\mathbf{eval}_{A'}$ is also defined (see Section 2.4) and in the case of a contracting $F$-magma $A$ for the same reasons (see Section 2.2). The notation $\mathbf{derop}_A$ will be used in these two extensions. These remarks can be summarized as follows.

**Proposition 3.2.3.** (1) *$M(F, V)$ is the free F-magma generated by $V$.*

(2) *$M^\infty(F, V)$ is the free contracting F-magma generated by $V$.*

(3) *$M_\Omega^\infty(F, V)$ is the free $\omega$-complete F-magma generated by $V$.*

Definition (3.2.2) uses an $(F \cup V_k)$-magma structure on $(A^k \to A)$, the set of total mappings: $A^k \to A$, with $v_i$ denoting the $i$th projection and $f$ denoting $\lambda g_1, g_2, \ldots, g_n \in (A^k \to A) \,.\, f_A \circ (g_1, \ldots, g_n)$. Hence it is based on the fact that $M(F, V_k) = M(F \cup V_k)$, the initial $(F \cup V_k)$-magma.

By defining $d'(\alpha, \alpha') = \mathbf{Sup}\{d(\alpha(a_1, \ldots, a_i), \alpha'(a_1, \ldots, a_k)) \mid a_1, \ldots, a_k \in A\}$ we make $(A^k \to A)$ into a contracting $(F \cup V_k)$-magma if $A$ is contracting.

By defining $\alpha \leqslant \alpha'$ iff $\alpha(a_1, \ldots, a_k) \leqslant \alpha'(a_1, \ldots, a_k)$ for all $a_1, \ldots, a_k$ we make it into an $\omega$-complete $F$-magma if $A$ is $\omega$-complete.

We can now state:

**Proposition 3.2.4.** (1) *If $A$ is an F-magma,* $\mathbf{derop}_A$ *is the unique $(F \cup V_k)$-homomorphism: $M(F, V_k) \to (A^k \to A)$.*

(2) *If $A$ is a contracting F-magma,* $\mathbf{derop}_A$ *is the unique $(F \cup V_k)$-homomorphism (of contracting magmas): $M^\infty(F, V_k) \to (A^k \to A)$.*

(3) *If $A$ is an $\omega$-complete F-magma,* $\mathbf{derop}_A$ *is the unique $\omega$-continuous $(F \cup V_k)$-homomorphism: $M_\Omega^\infty(F, V_k) \to (A^k \to A)$.*

**Proof.** (2) follows from the fact that $\mathbf{derop}_A : M^\infty(F, V_k) \to (A^k \to A)$ defined by (3.2.1) is uniformly continuous ($d(t, t') = (1/2)^n$ implies $d'(\mathbf{derop}_A(t), \mathbf{derop}_A(t')) \leq \|A\|^n$ and $\|A\| < 1$).

(3) follows from the fact that $\mathbf{derop}_A : M^\infty_\Omega(F, V_k) \to (A^k \to A)$ defined by (3.2.1) is monotone and $\omega$-continuous. □

### 3.3. First-order substitutions

By a *first-order substitution* we mean the operation which substitutes simultaneously a tree $\sigma(v)$ for each occurrence of a variable $v$ in a tree $t$ yielding a tree $\sigma(t)$.

We shall first give a direct definition of this operation in terms of trees defined as mappings from tree-domains to sets of symbols. Equivalent definitions will be given later, using the following general pattern:

- definition by structural induction for finite trees
- extension to infinite trees by uniform continuity or by $\omega$-continuity.

The following definitions will be given with respect to a ranked alphabet $F$, a finite or infinite set $V$ of variables such that $V \cap F = \emptyset$, a ranked alphabet $G$ that is not necessarily disjoint from $F$ and $V$.

**Definition.** Let $t \in M^\infty(F, V)$, let $\sigma(v)$ be a tree in $M^\infty(G)$ for all $v$ in $V$.

The result of the *simultaneous substitution* of $\sigma(v)$ *for* $v \in V$ *in* $t$ is the tree $t'$ defined as follows:

For all $\alpha$ in $\mathbf{N}_+^*$, $t'(\alpha)$ is defined if and only if: either $\alpha \in \mathbf{Dom}(t)$, $t(\alpha) \notin V$ and then $t'(\alpha) = t(\alpha)$ or $\alpha = \beta\alpha'$ for some $\beta \in \mathbf{Occ}(v, t)$, some $v$ in $V$, some $\alpha'$ in $\mathbf{Dom}(\sigma(v))$ and then $t(\alpha) = \sigma(v)(\alpha')$.

It can be checked that $t'$ is a perfectly well-defined tree in $M^\infty(F \cup G)$. We denote it by $\sigma(t)$; hence we consider $\sigma$ as extended from $V$ to $M^\infty(F, V)$.

Such a mapping $\sigma$ is called a *first-order substitution*. We shall also refer to *the first-order substitution* as the binary function associating $\sigma(t)$ with $t$ in $M^\infty(F, V)$ and $\sigma : V \to M^\infty(G)$, e.g. in Proposition 3.3.3.

We shall also use the notation $t[\sigma(v)/v; v \in V]$ for $\sigma(t)$.

In many cases, $V$ will be $\{v_{i_1}, \ldots, v_{i_k}\}$ and we shall use the notation $t[u_1/v_{i_1}, \ldots, u_k/v_{i_k}]$ with $u_j = \sigma(v_{i_j})$ for $1 \leq j \leq k$.

We shall also use the notation $t[u_1, \ldots, u_k]$ when $V = \{v_1, \ldots, v_k\}$ is known from the context.

**Remark 3.3.1.** It is easy to check from the definition that a first-order substitution $\sigma : M^\infty(F, V) \to M^\infty(F \cup G)$ satisfies the following properties:

(i) $\sigma(f) = f$ if $f \in F_0$,

(ii) $\sigma(f(t_1, \ldots, t_n)) = f(\sigma(t_1), \ldots, \sigma(t_n))$ if $f \in F_n$, $n \geq 1$, $t_1, \ldots, t_k \in M^\infty(F, V)$,

i.e. that $\sigma$ is an $F$-homomorphism.

Conversely, every $F$-homomorphism $\varphi : M(F, V) \to M^{\infty}(F \cup G)$ satisfies:

$$\varphi(t) = t[\varphi(v)/v ; v \in V],$$

i.e. is the restriction to $M(F, V)$ of a first-order substitution.

We shall characterize first-order substitutions as *continuous* or *$\omega$-continuous* $F$-homomorphisms.

Note also that (i) and (ii) above give a definition by structural induction of the extension of a mapping $\sigma : V \to M^{\infty}(G)$ into a mapping $\sigma : M(F, V) \to M^{\infty}(F \cup G)$. The extension to $M^{\infty}(F, V)$ will be made 'by continuity' (in two ways).

Let us finally remark that if $T \subset M^{\infty}(F)$, $G \subset F$ the set $M(G, T)$ defined above as the least sub-$G$-magma of $M^{\infty}(F)$ containing $T$ is

$$\{\sigma(t) \mid t \in M(G, V) \text{ and } \sigma(v) \in T \text{ for all } v \text{ in } V\}.$$

Let us define the distance of two substitutions $\sigma, \sigma' : V \to M^{\infty}(G)$ as $\mathbf{Sup}\{d(\sigma(v), \sigma'(v)) \mid v \in V\}$.

**Proposition 3.3.2.** *For all $t, t'$ in $M^{\infty}(F, V)$, all $\sigma, \sigma' : V \to M^{\infty}(G)$ we have*:

(1) $d(\sigma(t), \sigma'(t')) \leq \mathbf{Max}\{d(t, t'), d(\sigma, \sigma')\}$,

(2) $d(\sigma(t), \sigma'(t)) \leq (1/2) \,.\, d(\sigma, \sigma')$ *if* $t \notin V$.

**Proof.** We shall only prove (1). Just to simplify the notations and without loss of generality, we shall assume that $V = \{v_1, v_2, \ldots, v_k\}$. Hence $\sigma(t) = t[t_1, \ldots, t_k]$, $\sigma'(t') = t'[t'_1, \ldots, t'_k]$. Let $u = \sigma(t)$ and $u' = \sigma'(t')$.

Let $\alpha$ be a minimal element of $\mathbf{Dom}(u) \cap \mathbf{Dom}(u')$ such that $u(\alpha) \neq u'(\alpha)$. Then either $\alpha = \beta\gamma$, $\beta \in \mathbf{Occ}(v_i, t)$, $\gamma \in \mathbf{Dom}(t_i)$, $u(\alpha) = t_i(\gamma)$ or $\alpha \in \mathbf{Occ}(f, t)$ for some $f \in F$ and $u(\alpha) = f$.

A similar alternative holds for $\alpha$ with respect to $t', t'_1, \ldots, t'_k$. Let us only consider the case $\alpha = \beta\gamma = \beta'\gamma'$.

If $\beta \neq \beta'$ or $\beta = \beta'$ with $\beta \in \mathbf{Occ}(v_i, t)$, $\beta' \in \mathbf{Occ}(v_{i'}, t')$, $i \neq i'$ then $\delta(t, t') \leq \mathbf{Min}\{|\beta|, |\beta'|\}$ hence $\delta(u, u') \geq \delta(t, t')$, i.e. $d(u, u') \leq d(t, t')$. If $\beta = \beta'$, $\beta \in \mathbf{Occ}(v_i, t)$, $\beta' \in \mathbf{Occ}(v_i, t')$ then $\delta(u, u') = |\beta| + \delta(t_i, t'_i)$ hence $d(u, u') \leq d(t_i, t'_i) \leq d(\sigma, \sigma')$. □

Part (1) shows that the first-order substitution is uniformly continuous in all its arguments.

**Proposition 3.3.3.** *The first-order substitution as a mapping: $M^{\infty}_{\Omega}(F, V) \times (V \to M^{\infty}_{\Omega}(G)) \to M^{\infty}_{\Omega}(F \cup G)$ is $\omega$-continuous in all its arguments.*

The proof is omitted.

**Corollary 3.3.4.** *Let $A$ be a contracting (resp. $\omega$-complete) $F$-magma. For every $t$ in $M^{\infty}(F, V_k)$ (resp. in $M^{\infty}_{\Omega}(F, V_k)$) for every $t_1, \ldots, t_k$ in $M^{\infty}(F, V_n)$ (resp. in*

$M_\Omega^\infty(F, V_n)$),

$$\mathbf{derop}_A(t[t_1, \ldots, t_k]) = \mathbf{derop}_A(t) \circ (\mathbf{derop}_A(t_1), \ldots, \mathbf{derop}_A(t_k)).$$

**Proof.** For fixed $k$, $t_1, \ldots, t_k$, one can prove this for all $t$ in $M(F, V_k)$ (resp. in $M_\Omega(F, V_k)$) and then by continuity, this extends to all $t$ in $M^\infty(F, V_k)$ by Proposition 3.3.2 (resp. to all $t$ in $M_\Omega^\infty(F, V_k)$ by Proposition 3.3.3. □

**Proposition 3.3.5.** *The following properties of a mapping* $\sigma : M^\infty(F, V) \to M^\infty(F \cup G)$ *are equivalent*:

(1) $\sigma(t) = t[\sigma(v)/v;\ v \in V]$, *i.e.* $\sigma$ *is a first-order substitution*,

(2) $\sigma(t[t_1/v_1, \ldots, t_l/v_l]) = t[\sigma(t_1)/v_1, \ldots, \sigma(t_l)/v_l]$

*for all* $l \geqslant 0$, *all* $t$ *in* $M^\infty(F, V_l)$ *and* $t_1, \ldots, t_l$ *in* $M^\infty(F, V)$,

(3) $\sigma$ *is uniformly continuous and is an F-homomorphism*,

(4) $\sigma$ *is uniformly continuous and* $\sigma \restriction M(F, V)$ *is an F-homomorphism*.

*Furthermore, if* $\Omega \in F$, *they are equivalent to the following ones*:

(5) $\sigma$ *is* $\omega$*-continuous and is an F-homomorphism*,

(6) $\sigma$ *is* $\omega$*-continuous and* $\sigma \restriction M(F, V)$ *is an F-homomorphism*.

**Proof.** (1)⇒(2) follows from Proposition 3.4.2 given below.

(2)⇒(3). Proposition 3.3.2 shows that

$$d(\sigma(t), \sigma(t')) \leqslant d(t, t')$$

for all $t, t'$ in $M^\infty(F, V)$.

It suffices to take $t = f(v_1, \ldots, v_k)$ in (2) to see that $\sigma$ is an $F$-homomorphism. Note that (2) implies trivially the validity of (1) for all $t$ in $M(F, V)$.

(3)⇒(4). Trivially.

(4)⇒(1). The identity $\sigma(t) = t[\sigma(v)/v;\ v \in V]$ can be proved by structural induction on $t$, for all $t$ in $M(F, V)$. Since $\sigma$ is assumed uniformly continuous, it extends to all $t$ in $M^\infty(F, V)$.

(1)⇒(5). By Proposition 3.3.3.

(5)⇒(6). Trivially.

(6)⇒(1). As for (4)⇒(1) by $\omega$-continuity. □

### 3.4. *Miscellaneous properties of first-order substitutions*

In most proofs dealing with first-order substitution of trees, one need not go back to the definitions but one can just use a few properties.

All proofs will be omitted. They can be done directly from the definitions.

**Proposition 3.4.1.** *Let* $\sigma$, $\sigma'$ *be first-order substitutions*: $M^\infty(F, V) \to M^\infty(F \cup G)$. *Let* $s \in M^\infty(F, V)$.

(1) *If* $v \in \mathbf{Var}(s)$ *and* $\sigma(s) = \sigma'(s)$ *then* $\sigma(v) = \sigma'(v)$.

(2) *If* $v \notin \mathbf{Var}(s)$ *and* $V' = V - \{v\}$ *then* $\sigma(s) = s[\sigma(v')/v';\ v' \in V']$.

(3) $\mathbf{Subtree}(\sigma(s)) = \{\sigma(s') \mid s' \in \mathbf{Subtree}(s)\} \cup \{u \mid u \in \mathbf{Subtree}(\sigma(v)),\ v \in \mathbf{Var}(s)\}$.

We assume here that $G \cap V' = \emptyset$ and as above that $F \cap V = \emptyset$.

**Proposition 3.4.2** (Associativity). *Let* $\sigma : M^\infty(F, V) \to M^\infty(G, V')$ *and* $\theta : M^\infty(G, V') \to M^\infty(H)$ *be first-order substitutions. Then for all* $t$ *in* $M^\infty(F, V)$, $\theta(\sigma(t)) = (\theta \circ \sigma)(t) = \tau(t)$ *where* $\tau$ *is the first-order substitution*: $M^\infty(F - V', V \cup (V' \cap F)) \to M^\infty(H)$ *such that*

$$\tau(v) = \begin{cases} \theta(\sigma(v)) & \text{if } v \in V, \\ \theta(v) & \text{if } V' \cap F. \end{cases}$$

**Proof.** Notice that $\theta(v) = \theta(\sigma(v))$ if $v \in (V' \cap F) - V$. $\square$

Remark that if $V' \cap F = \emptyset$ (in particular if $V = V'$) then $\tau$ is the substitution: $M^\infty(F, V) \to M^\infty(H)$ such that $\tau(v) = \theta(\sigma(v))$ for $v$ in $V$. Another special case is the following:

**Proposition 3.4.3** (Commutativity). *If* $\sigma : M^\infty(F, V) \to M^\infty(F, V)$ *and* $\theta : M^\infty(F, V') \to M^\infty(F, V')$ *are first-order substitutions with* $V \cap V' = \emptyset$ *and* $F \cap (V \cup V') = \emptyset$ *then for all* $t$ *in* $M^\infty(F, V \cup V')$,

$$\sigma(\theta(t)) = \theta(\sigma(t)) = \tau(t)$$

*where* $\tau : M^\infty(F, V \cup V') \to M^\infty(F, V \cup V')$ *is the first-order substitution such that*

$$\tau(v) = \begin{cases} \sigma(v) & \text{if } v \in V, \\ \theta(v) & \text{if } v \in V'. \end{cases}$$

The following proposition describes the effect of first-order substitution on branch languages (c.f. Section 1.6).

Let us first remark that locally finite trees are preserved under first-order substitution, i.e. that $\sigma(t)$ is locally finite if $t$ and $\sigma(v)$ for all $v$ in $V$ are so.

**Proposition 3.4.4.** *Let* $t \in M^\infty(F, V)$ *and* $\sigma$ *be a first-order substitution*: $V \to M^\infty(G)$.

$$\mathbf{Brch}(\sigma(t)) = (\mathbf{Brch}(t) \cap \bar{F}^*) \cup \bigcup \{u\,\mathbf{Brch}(\sigma(v)) \mid u \in \bar{F}^*,\ uv \in \mathbf{Brch}(t),\ v \in V\},$$

$$\mathbf{PBrch}(\sigma(t)) = (\mathbf{PBrch}(t) \cap \bar{F}^*) \cup \bigcup \{u\,\mathbf{PBrch}(\sigma(v)) \mid u \in \bar{F}^*,\ uv \in \mathbf{Brch}(t),\ v \in V\}.$$

### *3.5. Second-order substitutions*

The second-order substitution consists in substituting trees for function symbols in trees.

Every first-order substitution can be viewed as a second-order one, but second-order substitutions are more difficult to study than first-order ones for the following reasons:

(1) The result of second-order substitutions cannot be easily defined as in the case of a first-order one (cf. Section 3.3). Hence we shall not define them 'directly' on all trees, but only on finite trees first, and this by structural induction.

(2) The extension to infinite trees does not always work in the metric approach due to a lack of continuity for certain *erasing* substitutions.

Let us note that this very problem occurs when one wants to define homomorphisms of infinite words [55].

**Definitions.** Let $F$ and $G$ be two ranked alphabets, not necessarily disjoint and let $V$ be a set of variables; $V \cap (F \cup G) = \emptyset$.

Let $t \in M(F, V)$, let $F'$ be a subset of $F$; for each $f$ in $F'$, let $\nu(f)$ be an element of $M^{\infty}(G, V_{\rho(f)})$.

The result of the (*simultaneous*) *substitution of* $\nu(f)$ *for* $f \in F'$ *in* $t$ is the tree $\theta(t)$ also denoted by $t\{\nu(f)/f; f \in F'\}$ and defined as follows by induction on the structure of $t$:

- if $t = f(t_1, \ldots, t_n)$ with $f \notin F'$ then $\theta(t) = f(\theta(t_1), \ldots, \theta(t_n))$,
- if $t = f(t_1, \ldots, t_n)$ then $\theta(t) = \nu(f)[\theta(t_1), \ldots, \theta(t_n)]$.

Hence $\theta$ is a mapping: $M(F) \to M^{\infty}(G \cup (F - F'))$. In order to simplify the notation, we shall assume that $F - F' \subseteq G$ in the sequel. Note that the variables of $V$ which appear in the $\nu(f)$'s do not appear in the images by $\theta$ of the elements of $M(F)$. Such a mapping is called a *second-order substitution*. (But as for first-order substitution, we shall also talk of second-order substitution as a binary mapping $M(F) \times (F' \to M^{\infty}(G, V)) \to M^{\infty}(G)$.) It extends to $M_{\Omega}(F)$ by means of the rule

$$\theta(\Omega) = \Omega.$$

We shall never substitute anything for $\Omega$, i.e. we shall never put $\Omega$ in $F'$.

A second-order substitution as above is *erasing* if $\nu(f) \in V$ for some $f$ in $F'$ and *nonerasing* otherwise. We say that $f$ such that $\nu(f) \in V$ is *erased*.

If $F' = \{f_1, \ldots, f_k\}$ we shall also use the notation $t\{\nu(f_1)/f_1, \ldots, \nu(f_k)/f_k\}$ for $t\{\nu(f)/f; f \in F'\}$ and the notation $t\{\nu(f_1), \ldots, \nu(f_k)\}$ if the sequence $f_1, \ldots, f_k$ is known from the context.

We shall compare two second-order substitutions $\theta$ and $\theta'$ associated with $\nu$ and $\nu'$ by:

$$\theta \leq \theta' \text{ if and only if } \nu(f) \leq \nu'(f) \text{ for all } f \text{ in } F',$$

$$d(\theta, \theta') = \mathbf{Sup}\{d(\nu(f), \nu'(f)) \mid f \in F'\}.$$

**Lemma 3.5.1.** (1) *The second-order substitution considered as a mapping*: $M_{\Omega}(F) \times (F' \to M_{\Omega}^{\infty}(G, V)) \to M_{\Omega}^{\infty}(G)$ *is monotone*; *it is* $\omega$*-continuous with respect to its second argument.*

(2) *Let $\theta$, $\theta'$ be second-order substitutions*: $M(F) \to M^{\infty}(G)$. *For all t in* $M(F)$,

$$d(\theta(t), \theta'(t)) \leq d(\theta, \theta').$$

(3) *If $\theta$ as above is not trivial, i.e. if $\theta(u) \neq \theta(u')$ for some u and u' in $M(F)$, the following conditions are equivalent*:

(i) $\|\theta\| \leq 1$,

(ii) *$\theta$ is uniformly continuous,*

(iii) *$\theta$ is nonerasing.*

**Proof.** We shall only prove (3). Let $\theta$ be nonerasing.

Let us show that for all $t$, $t'$ in $M(F, V)$,

$$\delta(t, t') \leq \delta(\theta(t), \theta(t')).$$

We show that for all $n$, for all $t, t' \in M(F, V)$,

$$n \leq \delta(t, t') \Rightarrow n \leq \delta(\theta(t), \theta(t')).$$

We do the proof by induction on $n$.

There is nothing to prove if $n = 0$.

Otherwise, let $n = n' + 1 \leq \delta(t, t')$. Then $t = f(t_1, \ldots, t_k)$ and $t' = f(t'_1, \ldots, t'_k)$ with $\delta(t_i, t'_i) \geq n'$ for $i = 1, \ldots, k$.

Hence $\delta(\theta(t_i), \theta(t'_i)) \geq n'$ for all $i = 1, \ldots, k$. Then

$$\delta(\theta(t), \theta(t')) = \delta(\nu(f)[\theta(t_1), \ldots, \theta(t_k)], \nu(f)[\theta(t'_1), \ldots, \theta(t'_k)])$$

$$\geq 1 + \mathbf{Min}\{\delta(\theta(t_i), \theta(t'_i)) \mid 1 \leq i \leq k\} \geq 1 + n' = n$$

by Proposition 3.3.2.

Hence

$$d(\theta(t), \theta(t')) \leq d(t, t')$$

which shows that $\theta$ is uniformly continuous and $\|\theta\| \leq 1$.

Conversely, let us assume that $\theta$ is erasing, i.e. without loss of generality that $\nu(f) = v_1$ for some $f$ of arity $\geq 1$.

Let us define $t_0 = u$, $t'_0 = u'$ and for all $n$, $t_{n+1} = f(t_n, t_n, \ldots, t_n)$, $t'_{n+1} = f(t'_n, \ldots, t'_n)$.

For all $n \geq 0$,

$$\theta(t_n) = \theta(u), \qquad \theta(t'_n) = \theta(u'), \qquad d(t_n, t'_n) \leq (1/2)^n.$$

It follows that $\|\theta\| = \infty$, hence that $\|\theta\| > 1$. Hence $\theta$ is not uniformly continuous. □

A second-order substitution $\theta : M_\Omega(F) \to M_\Omega^\infty(G)$ can be extended into $\theta : M_\Omega(F, V) \to M_\Omega^\infty(G, V)$ by the extra condition:

$$\theta(v) = v \quad \text{for all } v \text{ in } V.$$

**Lemma 3.5.2.** *A mapping $\theta : M_\Omega(F, V) \to M_\Omega^\infty(G, V)$ is a second-order substitution if and only if it satisfies the following conditions*:

(i) $\mathbf{Var}(\theta(t)) \subseteq \mathbf{Var}(t)$ *for all* $t$ *in* $M_\Omega(F, V)$,
(ii) $\theta(t) = t$ *for all* $t$ *in* $V \cup \{\Omega\}$,
(iii) $\theta(f(t_1, \ldots, t_l)) = \theta(f(v_1, \ldots, v_l))[\theta(t_1), \ldots, \theta(t_l)]$ *for all* $f$ *in* $F$, *all* $t_1, \ldots, t_l$ *in* $M_\Omega(F, V)$.

**Proof.** For the necessity, note that $\theta(f(v_1, \ldots, v_l)) = \nu(f)$ for $f \in F'$ if $\theta$ is a second-order substitution associated with $\nu$ and $F'$.

For the converse, it suffices to choose $F' = F$ and $\nu(f) = \theta(f(v_1, \ldots, v_l))$, $l = \rho(f)$ for all $f$ in $F$. □

**Notation.** From now on the mapping $\nu : F' \to M^\infty(G, V)$ which defines a second-order substitution: $M_\Omega(F, V) \to M_\Omega^\infty(G, V)$ will be extended to $F$ by $\nu(f) = f(v_1, \ldots, v_{\rho(f)})$ for $f$ in $F - F'$.

Lemma 3.5.1 shows that second-order substitutions can be extended from finite trees to infinite ones by $\omega$-continuity, i.e. by application of Proposition 2.4.2. Its second part shows that the extension using continuity, i.e. Proposition 2.2.1 can be made only for nonerasing substitutions.

We shall see that in the case of a nonerasing substitution the two extensions coincide on $M^\infty(F, V)$. In the case of an erasing substitution, the image of a tree in $M^\infty(F, V)$ can have occurrences of the symbol $\Omega$. The simplest example is

$$\nu(f) = v_1 \quad \text{where } \rho(f) = 1,$$

$$\theta(f^n\Omega) = \Omega \quad \text{for all } n \geq 0,$$

$$\theta(f^\omega) = \theta(\mathbf{Sup}(f^n\Omega)) = \mathbf{Sup}\,\theta(f^n\Omega) = \Omega.$$

By a *second-order substitution*: $M_\Omega^\infty(F, V) \to M_\Omega^\infty(G, V)$ we mean the extension by $\omega$-continuity of a second-order substitution: $M_\Omega(F, V) \to M_\Omega^\infty(G, V)$. A *weak second-order substitution* is a mapping: $M_\Omega^\infty(F, V) \to M_\Omega^\infty(G, V)$ (or: $M^\infty(F, V) \to M_\Omega^\infty(G, V)$) which satisfies conditions (i), (ii) and (iii) of Lemma 3.5.2 for all $t, t_1, \ldots, t_l$ in $M_\Omega^\infty(F, V)$ (resp. in $M^\infty(F, V)$). It is *erasing* if $\theta(f(v_1, \ldots, v_k)) \in V$ for some $f$ and *nonerasing* otherwise.

The following proposition is the analogous for second-order substitutions of Proposition 3.3.5.

**Proposition 3.5.3.** *A mapping is a second-order substitution if and only if it is an $\omega$-continuous weak second-order substitution.*

**Proof.** Immediate consequence of Lemma 3.5.2 and the various definitions. □

Here is a result showing that second-order substitutions are homomorphisms with respect to first-order substitution taken as an operation.

**Proposition 3.5.4.** *Let $\theta: M_\Omega^\infty(F, V) \to M_\Omega^\infty(G, V)$ be a second-order substitution. For every first-order substitution $\sigma: V \to M_\Omega^\infty(F, V)$ and every tree $t$ in $M_\Omega^\infty(F, V)$,*

$$\theta(\sigma(t)) = (\theta \circ \sigma)(t) = \tau(\theta(t))$$

*where $\tau$ is the first-order substitution such that $\tau(v) = \theta(\sigma(v))$ for all $v$ in $V$.*

In a special case and with another notation:

$$\theta(t[u_1, \ldots, u_k]) = \theta(t)[\theta(u_1), \ldots, \theta(u_k)].$$

**Proof.** This follows from Lemma 3.5.2 (ii) and (iii) for $t$ in $M_\Omega(F, V)$ by induction on the structure of $t$.

This identity extends to the case of $t$ in $M_\Omega^\infty(F, V)$ by $\omega$-continuity. □

**Example 3.5.5.** Here is a weak second-order substitution $\theta: M_\Omega^\infty(F, V) \to M_\Omega^\infty(G, V)$ which is not $\omega$-continuous.

$$F = \{f\}, \qquad \rho(f) = 1, \qquad G = \{a\}, \qquad \rho(a) = 0,$$

$$\nu(f) = v_1, \qquad \theta(f^n v_i) = v_i, \qquad \theta(f^n \Omega) = \Omega, \qquad \theta(f^\omega) = a.$$

It is not continuous either.

The following proposition extends to weak second-order substitutions some results of Lemma 3.5.1.

**Proposition 3.5.6.** *Let $\theta$ be a weak second-order substitution: $M^\infty(F, V) \to M^\infty(G, V)$ (not $M_\Omega^\infty(G, V)$). The following properties are equivalent:*

(1) *$\theta$ is nonerasing,*

(2) $\|\theta\| \leq 1$,

(3) *$\theta$ is uniformly continuous,*

(4) *$\theta$ is continuous,*

(5) *for every weak second-order substitution $\theta': M^\infty(F, V) \to M^\infty(G, V)$, if $\theta \restriction M(F, V) = \theta' \restriction M(F, V)$, then $\theta = \theta'$.*

**Proof.** (1) ⇒ (2). The proof given for the third part of Lemma 3.5.1 works for $t, t'$ in $M^\infty(F, V)$.

(2) ⇒ (3) and (3) ⇒ (4) are clear.

(4) ⇒ (1). Let $\theta$ be erasing. Without loss of generality we can assume that $\nu(f) = v_2$ and $\rho(f) = 2$.

Let $t$ be the infinite tree $f(v_1, f(v_1, f(v_1, \ldots)))$, let $u$ be $\theta(t)$, let $t_0$ be a variable such that $t_0 \neq u$ (hence $d(u, t_0) = 1$) and let $(t_n)$ be the sequence of trees such that $t_{n+1} = f(v_1, t_n)$ for $n \geq 0$.

It is clear that

$$t = \operatorname*{Lim}_{n\to\infty} t_n,$$

$$\theta(t_n) = \theta(t_0) = t_0 \quad \text{for all } n,$$

$$\theta(t) = u \neq t_0.$$

Hence $\theta$ is not continuous.

(5)$\Rightarrow$(1). Let $f$ and $t$ be as in the preceding proof.

We shall construct two weak second-order substitutions $\theta'$ and $\theta''$ such that $\theta'(t) \neq \theta''(t)$ and which coincide with $\theta$ on $M(F, V)$.

Let us assume that $F$ contains two constants $\Omega'$ and $\Omega''$ such that $\theta(\Omega') = \Omega'$ and $\theta(\Omega'') = \Omega''$ (otherwise we add them to $F$ or take variables instead). Letting them play the role of $\Omega$ in Section 2.4, we obtain two structures of $\omega$-complete $F$ magmas on $M^\infty(F, V)$ with respective partial orders $\leq'$ and $\leq''$.

Let $\theta'$ and $\theta''$ denote the canonical extensions of $\theta \upharpoonright M(F, V)$ to $M^\infty(F, V)$ with respect to $\leq'$ and $\leq''$. Proposition 3.5.3 shows that $\theta'$ and $\theta''$ are weak second-order substitutions and clearly, $\theta'(t) = \Omega'$, $\theta''(t) = \Omega'' \neq \Omega'$.

(1)$\Rightarrow$(5). Let $\theta$ be nonerasing and $\theta'$ be such that $\theta \upharpoonright M(F, V) = \theta' \upharpoonright M(F, V)$.

The substitution $\theta'$ is nonerasing too. Hence both of them satisfy (3) and $\theta = \theta'$ by continuity. □

**Corollary 3.5.7.** *If $\theta: M(F, V) \to M^\infty(G, V)$ is a nonerasing second-order substitution, its extension by $\omega$-continuity to $M_\Omega^\infty(F, V)$ and its extension by continuity to $M^\infty(F, V)$ coincide on $M^\infty(F, V)$.*

**Proof.** The mapping $\theta$ is uniformly continuous by Lemma 3.5.1, hence its extension by continuity to $M^\infty(F, V)$, let us denote it by $\bar{\theta}$, is well defined.

Let $\hat{\theta}$ be its extension by $\omega$-continuity to $M_\Omega^\infty(F, V)$.

It is uniformly continuous by Proposition 3.5.6 hence coincides with $\bar{\theta}$ on $M^\infty(F, V)$. □

**Remark 3.5.8.** A first-order substitution $\sigma: M^\infty(F, V) \to M^\infty(F \cup G)$ can be seen as the restriction to $M^\infty(F \cup V)$ of a nonerasing second-order substitution $\theta: M^\infty(F \cup V, X) \to M^\infty(F \cup G, X)$ defined by $\nu$ such that

$$\nu(v) = \sigma(v) \quad \text{for } v \text{ in } V,$$

$$\nu(f) = f(x_1, \ldots, x_n) \quad \text{for } f \text{ in } F_n,\ n \geq 0.$$

Note that we consider $V$ as a set of constants and we use another set of variables $X$ to define $\theta$.

The next proposition shows that the second-order substitution corresponds (semantically) to the replacement of a procedure name by the corresponding expression tree (or in terms of abstract data types, to implementing a data type by means of another one).

**Proposition 3.5.9.** *Let $\theta$ be a second-order substitution: $M(F, V)\to M(G, V)$ (resp. $M_\Omega^\infty(F, V)\to M_\Omega^\infty(G, V)$) (resp. $M^\infty(F, V)\to M^\infty(G, V)$ and nonerasing). Let $A = \langle D, (g_A)_{g\in G}\rangle$ be a G-magma (resp. an $\omega$-complete G-magma) (resp. a contracting G-magma), so that $f_B = \mathbf{derop}_A(\nu(f))$: $D^{\rho(f)}\to D$ is well-defined in the three cases. Let B be the F-magma $\langle D, (f_B)_{f\in F}\rangle$.*

*Then, for all t in $M(F, V_k)$, (resp. in $M_\Omega^\infty(F, V_k)$) (resp. in $M^\infty(F, V_k)$), $\mathbf{derop}_B(t)$ is defined and equal to $\mathbf{derop}_A(\theta(t))$.*

**Proof.** Note that $B$ is $\omega$-complete (resp. contracting) if $A$ is. Hence $\mathbf{derop}_B(t)$ and $\mathbf{derop}_A(\theta(t))$ are both defined in the three cases.

The equality is easy to prove for finite $t$'s and can be extended to infinite ones by $\omega$-continuity (resp. by continuity). □

*3.6. More on erasing substitutions*

Let $\theta$ be an erasing second-order substitution: $M(F, V)\to M^\infty(G, V)$, let $\hat\theta$ be its extension to $M_\Omega^\infty(F, V)$.

Our aim is to characterize the set of trees $t$ in $M^\infty(F, V)$ such that $\hat\theta(t)$ has no occurrence of $\Omega$.

Let us show with an example why this set is not always empty.

**Example 3.6.1.** Let us assume that $\rho(f) = 1$, $\rho(g) = 2$ and that

$$\nu(f) = v_1, \qquad \nu(g) = h(v_1).$$

Let $\theta$ be the second-order substitution: $M(F, V)\to M(H, V)$ associated with $\nu$, ($F = \{f, g\}$, $H = \{h\}$) and $\hat\theta$ be its extension to $M_\Omega^\infty(F, V)$. We have for example

$$\hat\theta(f^\omega) = \Omega, \qquad \hat\theta(g(v_1, f^\omega)) = h(v_1),$$

$$\hat\theta(g(g(g(\ldots, f^\omega), f^\omega), f^\omega)) = h^\omega, \qquad \hat\theta(g(f^\omega, f^\omega)) = h(\Omega).$$

It is clear that, for $t$ in $M^\infty(F, V)$ an occurrence of $\Omega$ in $\hat\theta(t)$ comes from a subtree of $t$ of the form $f^\omega$. And this subtree must not be in the scope of the second argument of some $g$ (if $t$ has a subtree $g(t_1, t_2)$ then $\hat\theta(t_2)$ does not contribute to $\hat\theta(t)$ since $v_2\notin \mathbf{Var}(\nu(g))$).

**Definition 3.6.2.** Let $\bar F$ be the alphabet $\{[f, i] \mid f\in F, 1\le i\le \rho(f)\}\cup F_0$ associated with $F$ as in Section 1.6.

Let $E_\theta = \{[f, i] \mid \rho(f)>0, \nu(f) = v_i\}$ ($E$ means 'erased') and let $N_\theta = \{[f, i] \mid \rho(f)>0, \nu(f)\notin V, v_i\in \mathbf{Var}(\nu(f))\}$ ($N$ means 'not erased').

We shall use the languages of branches $\mathbf{Brch}(t)$ and $\mathbf{Brch}^\omega(t)$ defined in Section 1.6.

If $t \in M_\Omega^\infty(F, V)$ then $\mathbf{Brch}(t) \subseteq \bar{F}_+^*(F_0 \cup V \cup \{\Omega\})$ (where $F_+ = \{f \in F \mid \rho(f) > 0\}$) and $\mathbf{Brch}^\omega(t) \subseteq \bar{F}_+^\omega$.

**Lemma 3.6.3.** *Let $t \in M_\Omega(F, V)$. Then $\theta(t) = v_i$ (resp. $v_i \in \mathbf{Var}(\theta(t))$) if and only if $wv_i \in \mathbf{Brch}(t)$ for some word $w$ in $E_\theta^*$ (resp. $w$ in $(E_\theta \cup N_\theta)^*$).*

Let us introduce

$$\mathbf{Brch}_\theta(t) = \mathbf{Brch}(t) \cap (E_\theta \cup N_\theta)^*(F_0 \cup V \cup \{\Omega\}),$$

$$\mathbf{Brch}_\theta^\omega(t) = \mathbf{Brch}^\omega(t) \cap (E_\theta \cup N_\theta)^\omega.$$

A tree $t$ in $M^\infty(F, V)$ is defined as *$\theta$-good* if $\mathbf{Brch}_\theta^\omega(t) \cap (E_\theta \cup N_\theta)^* E_\theta^\omega = \emptyset$.

We shall prove that a tree $t$ is $\theta$-good if and only if $\hat{\theta}(t)$ has no occurrence of $\Omega$.

Let us precise this new concept with some remarks and a lemma.

A finite tree is always $\theta$-good.

If an infinite tree is $\theta$-good its set of branches $\mathbf{Brch}_\theta(t) \cup \mathbf{Brch}_\theta^\omega(t)$ is included in $bN_\theta(E_\theta \cup N_\theta)^*(F_0 \cup V \cup \{\Omega\} \cup (E_\theta \cup N_\theta)^\omega)$ where $b$ is a word in $E_\theta^*$. This word $b$ is uniquely defined and defines the tree $u$ in the next lemma when $t$ is infinite.

**Lemma 3.6.4.** *A tree $t$ is $\theta$-good if and only if it can be written $t = u[t'_1, \ldots, t'_l]$ where*
*(i) $u \in M(F, V_l)$, $\theta(u) = v_1$,*
*(ii) $t'_1$ is finite or is of the form $f(t_1, \ldots, t_k)$ for $f$ in $F$ such that $\nu(f) \notin V$ and trees $t_1, \ldots, t_k$ such that $t_i$ is $\theta$-good for all $i$ in $[k]$ such that $v_i \in \mathbf{Var}(\nu(f))$.*

We can now state:

**Proposition 3.6.5.** *Let $\theta$ be a (possibly erasing) second-order substitution: $M(F, V) \to M^\infty(G, V)$. Let $\hat{\theta}$ denote its $\omega$-continuous extension: $M_\Omega^\infty(F, V) \to M_\Omega^\infty(G, V)$.*
*For all trees $t$ in $M^\infty(F, V)$, the following properties are equivalent:*
*(1) $t$ is $\theta$-good,*
*(2) $\hat{\theta}(t)$ has no occurrence of $\Omega$,*
*(3) $\hat{\theta}$ is continuous at $t$.*

**Proof.** (1) $\Rightarrow$ (2). Let $t$ be $\theta$-good.

If $t$ is finite then $\hat{\theta}(t) = \theta(t) \in M^\infty(G, V)$. If $t$ is infinite, we shall prove that $\|\hat{\theta}(t)\| = \infty$. We let $\|u\|$ denote $\mathbf{Min}\{|\alpha| \mid \alpha \in \mathbf{Occ}(\Omega, u)\}$.

Let $t$ be a $\theta$-good tree such $\|\hat{\theta}(t)\| < \infty$ and $\|\hat{\theta}(t)\|$ is minimal. We shall derive a contradiction.

Let $t = u[t'_1, \ldots, t'_l]$ as shown by Lemma 3.6.4. From Proposition 3.5.4 we get

$$\begin{aligned}\hat{\theta}(t) &= \theta(u)[\hat{\theta}(t'_1), \ldots, \hat{\theta}(t'_l)] \\ &= \hat{\theta}(t'_1) \quad \text{since } \theta(u) = v_1.\end{aligned}$$

If $t'_1$ is finite $\hat{\theta}(t'_1)$ has no occurrence of $\Omega$, hence $\|\theta(t)\| = \|\theta(t'_1)\| = \infty$, contradicting the initial assumption. Otherwise

$$t'_1 = f(t_1, \ldots, t_k) \quad \text{and} \quad \hat{\theta}(t'_1) = \nu(f)[\hat{\theta}(t_i)/v_i; i \in I]$$

where $I = \{i \mid v_i \in \mathbf{Var}(\nu(f))\}$. From this and since $\nu(f) \in M^\infty(G, V) - V$, if $\|\hat{\theta}(t')\| < \infty$, there must exist $i \in I$ such that $\|\hat{\theta}(t_i)\| < \infty$ and $\|\hat{\theta}(t_i)\| < \|\hat{\theta}(t)\|$.

Since $t_i$ is $\theta$-good, this contradicts the minimality of $\|\hat{\theta}(t)\|$.

(1)$\Rightarrow$(3). Let $t$ be $\theta$-good. Let $t_n$ be a sequence of trees in $M^\infty(F, V)$ which converges to $t$.

There exists an increasing sequence $u_n$ in $M_\Omega(F, V)$ such that

$$u_n \leq t_n \text{ and } u_n \leq t \quad \text{for all } n, \qquad t = \mathbf{Sup}_n (u_n).$$

It follows that

$$\hat{\theta}(u_n) \leq \hat{\theta}(t_n) \text{ and } \hat{\theta}(u_n) \leq \hat{\theta}(t) \quad \text{for all n},$$

$$\hat{\theta}(t) = \mathbf{Sup}_n\, \hat{\theta}(u_n),$$

$$\mathbf{Lim}_{n\to\infty} \|\hat{\theta}(u_n)\| = \infty \quad \text{since } \hat{\theta}(t) \text{ has no occurrence of } \Omega$$

$$\text{and } (\hat{\theta}(u_n))_{n\geq 0} \text{ is increasing.}$$

Since $\hat{\theta}(u_n) \leq \hat{\theta}(t_n)$ these trees "are equal on all levels less than $\|\hat{\theta}(u_n)\|$", i.e. $\delta(\hat{\theta}(u_n), \hat{\theta}(t_n)) \geq \|\hat{\theta}(u_n)\|$. Similarly, $\delta(\hat{\theta}(u_n), \hat{\theta}(t)) \geq \|\hat{\theta}(u_n)\|$ and finally $\delta(\hat{\theta}(t_n), \hat{\theta}(t)) \geq \|\hat{\theta}(u_n)\|$.

It follows that $d(\hat{\theta}(t_n), \hat{\theta}(t))$ converges to 0, i.e. that $\hat{\theta}(t) = \mathbf{Lim}_{n\to\infty}\, \hat{\theta}(t_n)$.

Hence $\hat{\theta}$ is continuous at $t$.

(2)$\Rightarrow$(1). We show that if $t$ is not $\theta$-good then $\hat{\theta}(t)$ contains an occurrence of $\Omega$. If $t$ is such that $\mathbf{Brch}^\omega_\theta(t) \subseteq E^\omega_\theta$ (in that case $\mathbf{Brch}^\omega_\theta(t)$ is reduced to a single infinite word) then $\hat{\theta}(t) = \Omega$.

Otherwise there exists in $\mathbf{Brch}^\omega_\theta(t)$ an infinite word of the form $bb_1$ with $b \in (E_\theta \cup N_\theta)^*$ and $b_1 \in E^\omega_\theta$. This shows that $t$ can be written $t = u[t_1, \ldots, t_l]$ for some $u$ in $M(F, V_l)$ in such a way that

$$bv_1 \in \mathbf{Brch}_\theta(u), \qquad \mathbf{Brch}^\omega_\theta(t_1) = \{b_1\}.$$

The tree $t$ is as in the special case we first considered, i.e. $\hat{\theta}(t_1) = \Omega$.

From Lemma 3.6.4, $v_1 \in \mathbf{Var}(\theta(u))$. Hence

$$\begin{aligned}\hat{\theta}(t) &= \theta(u)[\hat{\theta}(t_1), \ldots, \hat{\theta}(t_l)] \\ &= \theta(u)[\Omega, \hat{\theta}(t_2), \ldots, \hat{\theta}(t_l)]\end{aligned}$$

and contains occurrences of $\Omega$.

(3)$\Rightarrow$(1). We show that if $t$ is not $\theta$-good, $\hat{\theta}$ is not continuous at $t$.

Taking the notations of the preceding proof, let $(w_n)_{n\geq 0}$ be a sequence of finite trees such that

$$w_n = u[t_1^{(n)}, \ldots, t_l^{(n)}] \quad \text{for all } n \geq 0,$$

$$\mathbf{Lim}_{n\to\infty}(w_n) = t,$$

$$b_1^{(n)} v_k \in \mathbf{Brch}_\theta(t_1^{(n)}) \quad \text{for all } n \geq 0,$$

where $v_k$ is a variable which does not occur in $t$ and $b_1^{(n)}$ is the prefix of $b_1$ of length $n$.

Remark that

$$\theta(t_1^{(n)}) = v_k,$$

$$\hat{\theta}(w_n) = \theta(u)[\hat{\theta}(t_1^{(n)}), \ldots, \hat{\theta}(t_l^{(n)})]$$

$$= \theta(u)[v_k, \hat{\theta}(t_2^{(n)}), \ldots, \hat{\theta}(t_l^{(n)})].$$

Since $v_k$ does not occur in $t$, hence does not in $\hat{\theta}(t)$ either, $\delta(\hat{\theta}(w_n), \hat{\theta}(t)) \leq |\alpha|$ where $\alpha \in \mathbf{Occ}(v_k, \theta(u))$, i.e. $\hat{\theta}(w_n)$ does not converges to $\hat{\theta}(t)$. □

## *3.7. Miscellaneous properties of second-order substitutions*

A second-order substitution is *nondeleting* if $\mathbf{Var}(\nu(f)) = \{v_1, v_2, \ldots, v_k\}$ for all $k \geq 1$, all $f$ in $F_k$.

**Lemma 3.7.1.** *A second-order substitution is nondeleting if and only if* $\mathbf{Var}(\theta(t)) = \mathbf{Var}(t)$ *for all trees $t$ in $M(F, V)$ (resp. in $M_\Omega^\infty(F, V)$) (resp. in $M^\infty(F, V)$ when $\theta$ is non-erasing).*

**Proof.** As usual, by structural induction for finite trees and then, extension by continuity to infinite trees. □

Propositions 3.4.2 and 3.4.3 that we stated for first-order substitutions extend naturally to second-order ones.

Let us note in particular the following application of commutativity (i.e. the extension of Proposition 3.4.3):

**Corollary 3.7.2.** *Let $F \cap G = \emptyset$, let $t_1, \ldots, t_k \in M^\infty(G)$, let $f_1, \ldots, f_l \in F$ and let $u_i \in M^\infty(F \cup G, V_{\rho(f_i)}) - V$ for all $i = 1, \ldots, l$. Then, for all $s$ in $M^\infty(F \cup G, V_k)$:* $s[t_1/v_1, \ldots, t_k/v_k]\{u_1//f_1, \ldots, u_l//f_l\} = s\{u_1//f_1, \ldots, u_l//f_l\}[t_1/v_1, \ldots, t_k/v_k]$.

Part (3) of Proposition 3.4.1 extends as follows:

**Proposition 3.7.3.** $\mathbf{Subtree}(\theta(t)) = \mathbf{Var}(\theta(t)) \cup \{u[\theta(t_1), \ldots, \theta(t_n)] \mid u \in \mathbf{Subtree}(\nu(f)),$ $f(t_1, \ldots, t_n) \in \mathbf{Subtree}(t), f \in F_n, n \geq 0\}$.

**Proof.** Let us prove that every subtree $\theta(t)/w$ of $\theta(t)$ is of the form $v_i$ or $u[\theta(t_1), \ldots, \theta(t_n)]$ as required. The proof is an induction on $|w|$ and subsidiarly on the structure of $t$:

*Case* $|w|=0$: either $t=v_i$ or $t=f(t_1,\ldots,t_n)$ and we take $u=\nu(f)$;

*Case* $|w|>0$: then $t=f(t_1,\ldots,t_n)$. Either $w \in \mathbf{Dom}(\nu(f))$ and we are done with $u=\nu(f)/w$ or $w=w'w''$ with $w' \in \mathbf{Occ}(v_j, \nu(f))$, $w'' \in \mathbf{Dom}(t_j)$ and $|w''| \leq |w|$; then $\theta(t)/w=\theta(t_j)/w''$ and the induction hypothesis applied to $w''$ shows that $\theta(t)/w$ has the desired form if $|w''|<|w|$; if $|w''|=|w|$, i.e. if $|w'|=0$, the induction on the structure of $t$ can be used.

The converse inclusion is easier to prove. □

## 4. Regular trees

This section is devoted to *regular trees*. Such trees naturally arise in the process of 'unlooping' flowcharts. They also appear as results of *first-order unification* in the generalized sense of Huet [48].

We shall characterize regular trees as solutions in $M^{\infty}(F)$ of certain systems of equations. Solving such systems equation by equation will allow us to denote regular trees by some kind of *rational expressions* (Cousineau [29]). We shall also characterize them as forming the *free iterative theory generated by* $F$ (Ginali [40], Elgot et al. [34]). Finally we shall characterize them in terms of their language of branches (Courcelle [15, Section 1.6]) or their languages of occurrences (Ginali [40, Section 1.6]).

All definitions of this section will be given with respect to a fixed ranked alphabet $F$. The extension of all definitions and results to a sorted alphabet does not raise any difficulty except perhaps for notations. It will not be done.

### 4.1. Definitions

A tree $t$ is *regular* if the set **Subtree** $(t)$ of all its subtrees is finite. We shall denote by $R(F)$ the set of regular trees over $F$, i.e. in $M^{\infty}(F)$.

It is not difficult to establish the following properties:

$$M(F) \subsetneq R(F) \subsetneq M^{\infty}(F) \qquad (4.1.1)$$

(provided $F$ contains at least two symbols of arity $\geq 1$),

*$R(F)$ is closed under the F-operations.* (4.1.2)

*Any subtree of a regular tree is regular.* (4.1.3)

*The set of symbols occuring in a regular tree is finite.* (4.1.4)

We shall also use regular trees with variables. As for $M(F, V)$ and $M^{\infty}(F, V)$ we shall use the notation $R(F, V)$ for $R(F \cup V)$ in order to specify which symbols are variables i.e. are subject to substitution. The following fact is a straightforward consequence of Lemma 3.4.1:

*The family of regular trees is closed under first-order substitution.* (4.1.5)

**We mean by this that $\sigma(t)$ is regular if $t \in R(F, V)$ and $\sigma(v)$ is a regular tree for all $v$ in $V$. More generally:**

*The family of regular trees is closed under second-order substitution.* (4.1.6)

**We mean by this that $\theta(t)$ is regular if $t$ is regular and $\nu(f)$ is regular for all $f$ in $F$ (with the notations of Sections 3.5—3.7). This follows immediately from Proposition 3.7.3.**

*4.2. Systems of regular equations*

A *system of regular equations*, (we shall also say a *regular* system) is a finite system of the form $S = \langle x_1 = u_1, \ldots, x_n = u_n \rangle$ where $x_1, \ldots, x_n$ are the *unknowns* and $u_1, \ldots, u_n$ are elements of $F(\{x_1, \ldots, x_n\})$, i.e. are all of the form $f$ for $f$ in $F_0$ or $f(x_{i_1}, \ldots, x_{i_k})$ for $f$ in $F_k$, $k \geq 1$, $i_1, \ldots, i_k$ in $[n]$.

We associate with $S$ a mapping $|S| : M^\infty(F)^n \to M^\infty(F)^n$ defined by $|S|(t_1, \ldots, t_n) = (u_1[t_1/x_1, \ldots, t_n/x_n], \ldots, u_n[t_1/x_1, \ldots, t_n/x_n])$.

A *solution* of $S$ is an $n$-tuple $(t_1, \ldots, t_n)$ of trees in $M^\infty(F)$ *satisfying* the equations (where each $f$ in $F$ has its standard meaning on $M^\infty(F)$ (see Section 1), i.e. is a fix-point of the associated mapping $|S|$.

**Theorem 4.2.1.** *A regular system has a unique solution in $M^\infty(F)$. All components of this solution are regular trees. Every regular tree is a component of the unique solution of some regular system.*

**Proof.** The first two assertions will be proved later for more general systems of equations (Theorem 4.3.1).

Let $t$ be a regular tree. For each element $u$ of **Subtree**$(t)$ let us introduce an unknown $x_u$. Let $X$ be this set of unknowns. For each $x_u$ in $X$ we have to define an equation of the form $x_u = s_u$ for some $s_u$ in $F(X)$. If $u = f(u_1, \ldots, u_k)$ for $u_1, \ldots, u_k$ in **Subtree**$(t)$ we take $s_u = f(x_{u_1}, \ldots, x_{u_k})$. The system of all these equations is regular, the family of trees $(u)_{x_u \in X}$ is a solution of this system, hence its (unique) solution. The component of this solution corresponding to $x_t$ is clearly $t$. □

**Example 4.2.2.** Let $F = \{f, g, a, b\}$ with $\rho(f) = \rho(g) = 2$, $\rho(a) = \rho(b) = 0$.

The tree $t = f(a, g(b, g(b, g(b, \ldots))))$ is regular since **Subtree**$(t) = \{t, t_1, a, b\}$ where $t_1 = g(b, g(b, g(b, \ldots)))$. It is the first component of the unique solution in $M^\infty(F)$ of the regular system $S = \langle x_0 = f(x_2, x_1), x_1 = g(x_3, x_1), x_2 = a, x_3 = b \rangle$.

This tree is shown in Fig. 2.

*4.3. More general systems of regular equations*

A *generalized system of regular equations* (or a *generalized regular system*) is a finite system of the form $S = \langle x_1 = u_1, \ldots, x_n = u_n \rangle$ where $u_1, \ldots, u_n$ are elements of $M^{\infty}(F, X_n)$ and $X_n = \{x_1, \ldots, x_n\}$ is the set of *unknowns*. A mapping $|S|$ is associated with $S$ exactly as in Section 4.2.

A *solution* of $S$ is an $n$-tuple of trees $(t_1, \ldots, t_n)$ in $M^{\infty}(F)$ such that $t_i = u_i[t_1, \ldots, t_n]$, i.e. a fix-point of $|S|$.

If a system $S$ as above has regular right-hand sides, i.e. if $u_1, \ldots, u_n$ belong to $R(F, X)$ we say that $S$ is a system of *extended regular equations* or an *extended regular system*.

Finally, a generalized regular system $S$ satisfies the *Greibach condition* or is a *Greibach system* if none of its right-hand sides is an unknown.

**Theorem 4.3.1.** *A generalized regular Greibach system has a unique solution in $M^{\infty}(F)$. All components of the solution of an extended regular Greibach system are regular.*

**Proof.** Let $E = M^{\infty}(F)^n$ considered as a metric space. Then $E$ is complete since $M^{\infty}(F)$ is.

Since none of the $u_i$'s belongs to $\{x_1, \ldots, x_n\}$, the mapping $|S|$ is contracting. This follows from Proposition 3.3.2 part (2). Hence, by Theorem 2.1.1, $|S|$ has a unique fix-point, i.e. $S$ has a unique solution in $M^{\infty}(F)$.

Let us now assume that the $u_i$'s are regular. Let $(t_1, \ldots, t_n)$ be the solution of $S$.

Let $A = \bigcup\{A_j \mid 1 \leq j \leq n\}$ where $A_j = \{t'[t_1, \ldots, t_n] \mid t' \in \mathbf{Subtree}(u_j)\}$ for $j = 1, \ldots, n$.

Since the $u_j$'s are regular, the set $A$ is finite. Let us prove that $\mathbf{Subtree}(t_i) \subseteq A$ for all $i$ in $[n]$, and this will prove that the $t_i$'s are regular.

We show that for all $w$ in $\mathbf{N}_+^*$, for all $i$ in $[n]$ the tree $t_i/w$ belongs to $A$ if it is defined. And we do the proof by induction on $|w|$.

If $w \in \mathbf{Dom}(u_i)$ then $t_i/w = (u_i/w)[t_1, \ldots, t_n]$ hence belongs to $A_i$ since $u_i/w \in \mathbf{Subtree}(u_i)$.

If $w = w'w''$ for some occurrence $w'$ of $v_{i'}$ in $u_i$ and some node $w''$ of $t_{i'}$ (recall that $t_i = u_i[t_1, \ldots, t_n]$). Since $u_i \notin \{x_1, \ldots, x_n\}$ we have $|w'| > 0$ and $|w''| < |w|$, hence $t_i/w = t_{i'}/w''$ which belongs to $A$ by the induction hypothesis.

Otherwise $w$ does not belong to $\mathbf{Dom}(u_i)$ and there is nothing to prove. □

**Remark 4.3.2.** Let us define a generalized system as *proper* if it has a unique solution.

In order to characterize the proper systems let us say that an unknown $x_i$ of $S$ (as above) is *singular* if $|S|^p(x_1, \ldots, x_n) = (s_1, \ldots, s_n)$ with $s_i = x_i$ for some $p \geq 1$ (such a $p$ can be taken less than $n$ if it exists). Otherwise $x_i$ is *nonsingular*.

It has been shown by Bloom et al. [7] that a system is proper if and only if it has no singular unknown.

Two singular unknowns $x_i$ and $x_j$ are *independent* if for all $p \geq 1$, $x_i \neq x_j$ where as above, $|S|^p(x_1, \ldots, x_n) = (s_1, \ldots, s_n)$.

All solutions of a nonproper system can be defined parametrically in a unique way in terms of arbitrary values given to independent singular unknowns. More details will be given in Lemma 4.9.8.

### *4.4. Solving regular systems equation by equation*

Let $S = \langle x_1 = u_1, \ldots, x_n = u_n \rangle$ be a proper generalized regular system. Let us single out its first equation. It can be solved in $M^\infty(F, \{x_2, \ldots, x_n\})$ by considering the unknowns $x_2, \ldots, x_n$ as constants. Let $t_1$ denote its solution, i.e. the unique tree such that $t_1 = u_1[t_1/x_1]$.

Let now $S'$ be the system $\langle x_2 = u'_2, \ldots, x_n = u'_n \rangle$ where $u'_i = u_i[t_1/x_1]$ for $i = 2, 3, \ldots, n$. Let $(t'_2, \ldots, t'_n)$ be its solution.

**Claim 4.4.1.** (1) *The system $S'$ has a unique solution* $(t'_2, \ldots, t'_n)$.

(2) *The solution of $S$ is the n-tuple* $(t'_1, \ldots, t'_n)$ *where* $t'_1 = t_1[t'_2/x_2, \ldots, t'_n/x_n]$.

**Proof.** Let us verify that for any solution $(t'_2, \ldots, t'_n)$ the associated $(t'_1, t'_2, \ldots, t'_n)$ is a solution of $S$:

$$\begin{aligned} t'_1 &= t_1[t'_2/x_2, \ldots, t'_n/x_n] \\ &= u_1[t_1/x_1][t'_2/x_2, \ldots, t'_n/x_n] \\ &= u_1[t_1[t'_2/x_2, \ldots, t'_n/x_n]/x_1, t'_2/x_2, \ldots, t'_n/x_n] \\ &= u_1[t'_1/x_1, t'_2/x_2, \ldots, t'_n/x_n]. \end{aligned}$$

For $i = 2, \ldots, n$,

$$\begin{aligned} t'_i &= u'_i[t'_2/x_2, \ldots, t'_n/x_n] \\ &= u_i[t_1/x_1][t'_2/x_2, \ldots, t'_n/x_n] \\ &= u_i[t'_1/x_1, \ldots, t'_n/x_n] \end{aligned}$$

with similar computations as for $t'_1$.

Hence if $S'$ had two distinct solutions, so would have $S$ which is not the case by assumption. Hence $S'$ has a unique solution and it satisfies part (2) of the claim. □

Arguing by induction on the number of equations, one can show that solving a system of $n$ equations reduces to solving $n$ single equations and composing appropriately their solutions.

This method is fully similar to the one used in language theory to solve regular systems of equations in terms of rational expressions. This suggests to do the same for regular trees.

### *4.5. Rational expressions denoting regular trees*

Cousineau has defined in [29] a class of 'rational expressions' in order to denote regular trees obtained as solutions of regular systems. Our presentation of his results differs substantially from his.

**Definition 4.5.1.** We introduce on $M^\infty(F, V_k)$ a new operation named **Star**. For $t$ in $M^\infty(F, V_k)-\{v_1\}$ we define **Star**$(t)$ as the unique tree in $M^\infty(F, V_{k-1})$ such that

$$\textbf{Star}(t) = t[\textbf{Star}(t)/v_1, v_1/v_2, \ldots, v_{k-1}/v_k].$$

The existence and unicity of **Star**$(t)$ follows from Theorem 4.3.1.

If $t \in M^\infty(F, \{v_1\})$ then **Star**$(t) \in M^\infty(F)$ $(=M^\infty(F, V_0))$ and $t \in M^\infty(F)$ if and only if **Star**$(t) = t$.

Note that **Star**$(v_1)$ is not defined. We could define it as $\Omega$, the 'bottom' tree (see Section 3) and this would be useful for expressing *least* solutions of possibly nonproper regular system ($\Omega$ is clearly the least solution of the equation $x_1 = x_1$). We shall discuss this later (see Section 4.10) but we restrict here our attention to extended regular systems that are proper.

Remark finally that the star operation depends on a precise set of variables, here $V = \{v_1, v_2, \ldots, v_n, \ldots\}$ which will be kept fixed in this section.

**Lemma 4.5.2.** *Let* $u \in M^\infty(F, V_{k+1})-\{v_1\}$ *and* $t_1, \ldots, t_k \in M^\infty(F, V_l)$. *Then* **Star**$(u)[t_1, \ldots, t_k] =$ **Star**$(u[t'_1/v_2, \ldots, t'_k/v_{k+1}])$ *where* $t'_i = t_i[v_2/v_1, \ldots, v_{l+1}/v_l]$ *for* $i = 1, \ldots, k$.

**Proof.** We have **Star**$(u) = u[\textbf{Star}(u)/v_1, v_1/v_2, \ldots, v_k/v_{k+1}]$. Let $u' =$ **Star**$(u)[t_1, \ldots, t_k]$. Proposition 3.4.2 gives us

$$u' = u[u'/v_1, t_1/v_2, \ldots, t_k/v_{k+1}]. \tag{1}$$

On the other hand, **Star**$(u[t'_1/v_2, \ldots, t'_k/v_{k+1}])$ is the unique tree $w$ in $M^\infty(F, V_l)$ such that

$$w = u[t'_1/v_2, \ldots, t'_k/v_{k+1}][w/v_1, v_1/v_2, \ldots, v_l/v_{l+1}]. \tag{2}$$

Since $t'_1, \ldots, t'_k$ have no occurrence of $v_1$, (2) can be written

$$w = u[w/v_1, t''_1/v_2, \ldots, t''_k/v_{k+1}] \tag{3}$$

where

$$t''_i = t'_i[v_1/v_2, \ldots, v_l/v_{l+1}] \quad \text{for } i = 1, \ldots, k.$$

By definition of $t'_i$,

$$t''_i = t_i[v_2/v_1, \ldots, v_{l+1}/v_l][v_1/v_2, \ldots, v_l/v_{l+1}] = t_i. \tag{4}$$

Since $u \neq v_1$, the equation

$$x = u[x/v_1, t_1/v_2, \ldots, t_k/v_{k+1}]$$

has a unique solution in $M^\infty(F, V_l)$. This shows together with (1), (3) and (4) that $u' = w$. □

**Definition 4.5.3.** A *rational expression* is (in this paper) an element $e$ (or $e_1, e' \cdots$) of $M(F \cup \{*\}, V)$. The tree **Val**$(e)$ it possibly denotes can be inductively defined as follows:
- if $e = v_i$ then **Val**$(e) = v_i$,
- if $e = f(e_1, \ldots, e_k)$ then **Val**$(e) = f($**Val**$(e_1), \ldots,$ **Val**$(e_k))$ if each of **Val**$(e_1), \ldots,$ **Val**$(e_k)$ is defined and **Val**$(e)$ is undefined otherwise,
- if $e = *(e')$ then **Val**$(e) =$ **Star**$($**Val**$(e'))$ if **Val**$(e')$ is defined and is not $v_1$ and **Val**$(e)$ is undefined otherwise.

We say that $e$ is *defined* (*undefined*) if **Val**$(e)$ is.

It is easy to check that $e$ is undefined if and only if it contains a subexpression of the form $*(*(\cdots *(v_k))\cdots))$ with $k$ occurrences of $*$.

We can already solve some regular systems: for instance let $S = \langle x_1 = f(x_1, g), x_2 = h(x_1, x_2)\rangle$. Its solution is $(e_1, e_2)$, i.e. more precisely the pair (**Val**$(e_1)$, **Val**$(e_2)$) where $e_1 = *(f(v_1, g))$ and $e_2 = *(h(*(f(v_1, g)), v_1))$.

In order to apply the method of Section 4.4 and solve arbitrary systems we need a way to form a rational expression **Comp**$(e, e_1, \ldots, e_k)$ having the value **Val**$(e)$ [**Val**$(e_1), \ldots,$ **Val**$(e_k)$]. It is easy to check that taking $e[e_1/v_1, \ldots, e_k/v_k]$ would be incorrect.

**Definition 4.5.4.** Let **Comp**$(e, e_1, \ldots, e_k)$ be the rational expression defined as follows by induction on the structure of $e$:
- if $e = v_i$ and $1 \leqslant i \leqslant k$ then **Comp**$(e, e_1, \ldots, e_k) = e_i$,
- if $e = v_i$ and $i > k$ then **Comp**$(e, e_1, \ldots, e_k) = v_i$,
- if $e = f(e'_1, \ldots, e'_l)$ then **Comp**$(e, e_1, \ldots, e_k) = f(e''_1, \ldots, e''_l)$ where $e''_i =$ **Comp**$(e'_i, e_1, \ldots, e_k)$ for $i = 1, \ldots, l$
- if $e = *(e')$ then **Comp**$(e, e_1, \ldots, e_k) = *($**Comp**$(e', v_1, e''_1, \ldots, e''_k))$ where $e''_i =$ **Comp**$(e_i, v_2, v_3, \ldots, v_{l+1})$ for all $i = 1, \ldots, k$ and $l$ is large enough such that **Val**$(e_i) \in M^\infty(F, V_l)$ for all $i = 1, \ldots, k$.

In the last clause above, we shall put **Shift**$(e_i)$ instead of $e''_i$ where **Shift** is a mapping on regular expressions acting as $\lambda e\,.$ **Comp**$(e, v_2, v_3, \ldots, v_{l+1})$ but which can be defined directly.

Actually, we shall define **Shift**$(k, e)$ for $k \geqslant 1$ and we shall take

$$\mathbf{Shift}(e) = \mathbf{Shift}(1, e),$$

$$\mathbf{Shift}(k, v_i) = \begin{cases} v_i & \text{if } i < k, \\ v_{i+1} & \text{if } i \geqslant k, \end{cases}$$

$$\mathbf{Shift}(k, f(e_1, \ldots, e_l)) = f(\mathbf{Shift}(k, e_1), \ldots, \mathbf{Shift}(k, e_l)),$$

$$\mathbf{Shift}(k, *(e)) = *(\mathbf{Shift}(k+1, e)).$$

**Claim 4.5.5.** *For all defined rational expression $e$ and all integer $k \geqslant 1$,* **Val**$($**Shift**$(k, e)) =$ **Val**$(e)[v_{k+1}/v_k, v_{k+2}/v_{k+1}, \ldots, v_{l+1}/v_l]$ *where $l$ is such that* **Val**$(e) \in M^\infty(F, V_l)$.

**Proof.** By induction on the structure of $e$. We only consider the case $e = *(e')$ where $\mathbf{Val}(e') \in M^{\infty}(F, V_{l+1})$. Then

$$\begin{aligned}
&\mathbf{Val}(e)[v_{k+1}/v_k, \ldots, v_{l+1}/v_l] \\
&\quad = \mathbf{Star}(\mathbf{Val}(e'))[v_{k+1}/v_k, \ldots, v_{l+1}/v_l] \\
&\quad = \mathbf{Star}(\mathbf{Val}(e'))[v_1, v_2, \ldots, v_{k-1}, v_{k+1}, \ldots, v_{l+1}] \\
&\quad = \mathbf{Star}(\mathbf{Val}(e')[v_2/v_2, \ldots, v_k/v_k, v_{k+2}/v_{k+1}, \ldots, v_{l+2}/v_{l+1}]) \qquad (1) \\
&\quad = \mathbf{Star}(\mathbf{Val}(e')[v_{k+2}/v_{k+1}, \ldots, v_{l+2}/v_{l+1}]) \\
&\quad = \mathbf{Val}(*(\mathbf{Shift}(k+1, e'))) \qquad (2) \\
&\quad = \mathbf{Val}(\mathbf{Shift}(k, e)).
\end{aligned}$$

We have used Lemma 4.5.2 to obtain (1) and the induction hypothesis to obtain (2). □

**Claim 4.5.6.** *If* $e, e_1, \ldots, e_k$ *are defined rational expressions then* $\mathbf{Val}(\mathbf{Comp}(e, e_1, \ldots, e_k)) = \mathbf{Val}(e)[\mathbf{Val}(e_1), \ldots, \mathbf{Val}(e_k)]$.

**Proof.** By induction on the structure of $e$. Once again the only interesting case is $e = *(e')$. Then

$$\begin{aligned}
&\mathbf{Val}(\mathbf{Comp}(e, e_1, \ldots, e_k)) \\
&\quad = \mathbf{Val}(*(\mathbf{Comp}(e', v_1, \mathbf{Shift}(e_1), \ldots, \mathbf{Shift}(e_k)))) \\
&\quad = \mathbf{Star}(\mathbf{Val}(e')[t'_1/v_2, \ldots, t'_k/v_{k+1}])
\end{aligned}$$

by induction and Claim 4.5.5, with $t'_i = \mathbf{Val}(e_i)[v_2/v_1, \ldots, v_{l+1}/v_l]$. Hence, by Lemma 4.5.2,

$$\begin{aligned}
\mathbf{Val}(\mathbf{Comp}(e, e_1, \ldots, e_k)) &= \mathbf{Star}(\mathbf{Val}(e'))[\mathbf{Val}(e_1), \ldots, \mathbf{Val}(e_k)] \\
&= \mathbf{Val}(e)[\mathbf{Val}(e_1), \ldots, \mathbf{Val}(e_k)]. \quad \square
\end{aligned}$$

**Theorem 4.5.7.** *A tree is regular if and only if it is the value of a rational expression. For any proper extended regular system of equations, one can find rational expressions defining its solution.*

**Proof.** The value of a rational expression is a regular tree: this is an easy consequence of Theorem 4.3.1 and the definition of **Star**.

Conversely, let $S = \langle x_1 = u_1, \ldots, x_n = u_n \rangle$ be a regular system. We shall construct an $n$-tuple of rational expressions denoting its solution.

Actually, we shall do the construction in a more general case, where $S$ is a proper extended regular system where $u_i = \mathbf{Val}(e_i)[x_1/v_1, \ldots, x_n/v_n]$ for some rational expressions $e_1, \ldots, e_n$. (It is useful *not* to identify $x_i$ and $v_i$ as it will appear soon).

If $n = 1$ then the solution of $S$ is $t_1 = \mathbf{Val}(*(e_1))$.

Otherwise, we start solving $S$ by following the method of Section 4.4 (and using the same notations). It is clear that

$$t_1 = \mathbf{Val}(*(e_1))[x_2/v_1, \ldots, x_n/v_{n-1}].$$

In both cases $*(e_1)$ is defined since otherwise $\mathbf{Val}(e_1)$ would be $v_1$ and $S$ would not be proper.

We let $e_i' = \mathbf{Comp}(e_i, *(e_1), v_1, \ldots, v_{n-1})$ for $i = 2, \ldots, n$ so that the system $S'$ of Section 4.5 is exactly $\langle x_2 = u_2', \ldots, x_n = u_n' \rangle$ with $u_i' = \mathbf{Val}(e_i')[x_2/v_1, \ldots, x_n/v_{n-1}]$ for $i = 2, \ldots, n$.

By induction, we can assume that we know rational expressions $e_2'', \ldots, e_n''$ defining the solution of $S'$ and we need only compute $e_1'' = \mathbf{Comp}(e_1, e_2'', \ldots, e_n'')$ to obtain an $n$-tuple $(e_1'', e_2'', \ldots, e_n'')$ defining the solution of $S$. That $t_i' = \mathbf{Val}(e_i'')$ follows from Claim 4.4.1 and the induction hypothesis for $i = 2, \ldots, n$ and from Claims 4.4.1 and 4.5.6 for $i = 1$. □

**Example 4.5.8.** Let $S$ be the system

$$x = f(x), \qquad y = g(x, y, z), \qquad z = h(y, z).$$

Solving the first equation gives us

$$x = *(f(v_1))$$

(for simplicity, we identify a rational expression with its value). Then the system $S$ reduces to the following two equations:

$$y = g(*(f(v_1)), v_1, v_2)[y/v_1, z/v_2],$$

$$z = h(v_1, v_2)[y/v_1, z/v_2].$$

By defining $e$ as $g(*(f(v_1)), v_1, v_2)$, we get

$$y = *(e)[z/v_1]$$

and we are reduced to solve

$$z = h(*(e), v_1)[z/v_1].$$

We now obtain the final expressions for $z$ and $y$:

$$z = *(h(*(g(*(f(v_1)), v_1, v_2)), v_1)),$$

$$y = \mathbf{Comp}(*(e), e')$$

where $e'$ is the rational expression defining $z$. After evaluation of $\mathbf{Comp}(*(e), e')$ one gets

$$y = *(g(*(f(v_1)), v_1, *(h(*(g(*(f(v_1)), v_1, v_2)), v_1))).$$

*4.5.9. Applications to program transformations*

It is known that arbitrary flowcharts cannot be transformed into equivalent while-programs without introducing auxiliary variables (see for instance Elgot [33]). But such a transformation can be done if one allows (like in EXEL [56]) **do-repeat** loops with exit statements of the form **exit** $i$ for $i \geq 1$ (causing a jump out of the $i$th surrounding **do-repeat** loop). This can be established as a corollary of Theorem 4.5.7 in the following way.

From a rational expression denoting the execution tree of a flowchart, one can obtain a 'structured program scheme' by translating $*(\ldots)$ into **do**(...) **repeat**, $v_{i+1}$ into **exit** $i$ for $i \geq 1$, $v_1$ into the **null** statement (since $v_1$ corresponds to a return to the beginning of the surrounding **do**... **repeat** loop or to the **end** of the whole program). By replacing the action symbols by their meanings, written as sequences of ground statements, one obtains a program equivalent to the initial 'unstructured' program. This proof method is due to Cousineau [28].

*4.6. Iterative theories of trees*

We shall present a nice algebraic structure that one can put on $M^{\infty}(F, V)$ the basic operations of which are the *composition* (i.e. the first-order substitution) and the *iteration* (i.e. taking the solution of generalized regular system). It has been invented by Elgot [32] and developed in a series of papers [6, 7, 8, 10, 34, 40].

The basic objects will not be trees but $n$-tuples of trees also called *$n$-trees*. The reader will have noted that our substitution $t[u_1, \ldots, u_n]$ is a binary operation concerning a tree $t$ and an $n$-tree $u = (u_1, \ldots, u_n)$. Hence, the introduction of $n$-trees is natural in a theory emphasizing the properties of substitution whereas trees are natural in a theory emphasizing the $F$-operations.

Rather than starting with the general definition of an iterative theory, we describe the iterative theory $T$ 'of' infinite trees over the ranked alphabet $F$. The iterative theory $R$ 'of' regular trees over $F$ can be characterized as the *free iterative theory generated by $F$* as we shall see later.

The set of variables $V = \{v_1, \ldots, v_n, \ldots\}$ is fixed and will play a similar role as in Section 4.5.

For all integers $n, p \geq 0$ let us denote by $T_{n,p}$ the set of $n$-tuples of trees in $M^{\infty}(F, V_p)$, hereafter called *$n$-trees*. If $n = 0$ then $T_{n,p}$ is reduced to the empty sequence, here denoted by $0_p$. If $p = 0$ then $T_{n,p}$ consists of sequences of trees in $M^{\infty}(F)$ (unless $n = 0$).

The substitution extends in an obvious way into an operation associating with $s = (s_1, \ldots, s_n)$ in $T_{n,p}$ and $t = (t_1, \ldots, t_p)$ in $T_{p,q}$ the $n$-tree $s \cdot t = u = (u_1, \ldots, u_n)$ in $T_{n,q}$ such that $u_i = s_i[t_1/v_1, \ldots, t_p/v_p]$ for $i = 1, \ldots, n$.

Sequences can be formed by means of *tupling* which associates with $t_1, \ldots, t_p$ in $T_{1,n}$ the element $\langle t_1, \ldots, t_p \rangle$ of $T_{p,n}$ equal to $(u_1, \ldots, u_p)$ where $t_i = (u_i)$ and $u_i \in M^{\infty}(F, V_n)$ for $i = 1, \ldots, p$.

For all $p \geq 1$, we denote by $I_p$ the element $(v_1, \ldots, v_p)$ of $T_{p,p}$ and by $\pi_{i,p}$ the element $(v_i)$ of $T_{1,p}$ (for $1 \leq i \leq p$). These notations as justified by properties (4.6.2) and (4.6.3) below. The index $p$ will be frequently omitted.

The operations introduced above satisfy the following properties:

$$(s.t).u = s.(t.u), \tag{4.6.1}$$

$$s.I = s, \tag{4.6.2}$$

$$\pi_i.(t_1, \ldots, t_n) = t_i, \tag{4.6.3}$$

$$t = (\pi_1.t, \pi_2.t, \ldots, \pi_n.t) \tag{4.6.4}$$

for all objects $s$, $t$, $u$ of appropriate type.

Finally, for $t$ in $T_{n,q}$ and $u$ in $T_{p,q}$, we shall use $(t, u)$ as a shorthand for $(\pi_1.t, \pi_2.t, \ldots, \pi_n.t, \pi_1.u, \ldots, \pi_p.u)$. Hence $(t, u)$ is in $T_{n+p,q}$.

Up to now, we have only defined an *algebraic theory* $T = (T_{n,p})_{n,p \geq 0}$, and we already know two subtheories of $T$:

$$R = (R_{n,p})_{n,p \geq 0}, \quad \text{where } R_{n,p} = R(F, V_p)^n$$

and

$$M = (M_{n,p})_{n,p \geq 0}, \quad \text{where } M_{n,p} = M(F, V_p)^n.$$

We denote them by $T_F$, $R_F$ and $M_F$ if we want to indicate the alphabet.

We can reformulate Theorem 4.3.1 in the framework of the above algebraic theories.

Let $S = \langle v_1 = u_1, \ldots, v_n = u_n \rangle$ be a generalized regular system in Greibach normal form. Solving it amounts to finding $(t_1, \ldots, t_n)$ in $M^\infty(F)^n = T_{n,0}$ such that $t_i = u_i[t_1/v_1, \ldots, t_n/v_n]$ i.e. to finding some $t$ in $T_{n,0}$ such that

$$t = u.t \tag{4.6.5}$$

where $u$ is the element $(u_1, \ldots, u_n)$ of $T_{n,n}$.

Our Theorem 4.3.1 asserts the existence and unicity of such a $t$. It will be denoted by $u^\dagger$.

More generally all equations of the form

$$t = u.(t, I_p) \tag{4.6.6}$$

where $u \in T_{n,n+p}$, $\pi_i.u \neq \pi_j$ for all $i, j \in [n]$ have a unique solution in $T_{n,p}$, also denoted by $u^\dagger$ and which is the solution in $M^\infty(F, V_p)^n$ of the generalized regular system:

$$S = \langle x_1 = u'_1, \ldots, x_n = u'_n \rangle$$

where

$$u'_i = u_i[x_1/v_1, \ldots, x_n/v_n, v_1/v_{n+1}, \ldots, v_p/v_{n+p}] \quad \text{for } i = 1, \ldots, n.$$

We express this by saying that $T$ is *closed under conditional iteration* or is *iterative*.

Note that (4.6.5) is the special case of (4.6.6) where $p=0$.

Note also that for $u$ in $T_{1,n}$ identified with $M^{\infty}(F, V_n)$, $u^{\dagger}=\mathbf{Star}(u)$.

Theorem 4.3.1 also shows that $R$ is iterative whereas $M$ is not since the equation $x=f(x)$ has no solution in $M(F)$.

**Remark 4.6.7.** A similar structure has been proposed by Arnold and Dauchet [1] under the name of *magmoid.* The basic operations are the composition and the *tensor product* which associates with $u$ in $T_{n,p}$ and $u'$ in $T_{n',p'}$ the element $t=(t_1,\ldots,t_{n+n'})$ of $T_{n+n',p+p'}$, denoted by $t=u\otimes u'$, such that

$$t_i=u_i \quad \text{if } 1\leq i\leq n,$$

$$t_{i+n}=u'_i[v_{p+1}/v_1,\ldots,v_{p+p'}/v_{p'}] \quad \text{if } 1\leq i\leq n'.$$

There is no special notation for **Star** or $\dagger$.

## 4.7. *General iterative theories*

An *algebraic theory* $J$ consists of non-empty sets $J_{n,p}$ for $n,p\geq 0$ together with an operation named *composition* denoted by $\cdot$, a multi-adic operation named *source-tupling* and denoted by $(\,,\ldots,\,)$, objects $O_p$ in $J_{0,p}$ for $p\geq 0$, $I_p$ in $J_{p,p}$ and $\pi_{i,p}$ in $J_{1,p}$ for $1\leq i\leq p$ (also denoted by $I$ and $\pi_i$). The objects $\pi_i$ are said *distinguished.* All these objects and operations must satisfy conditions (4.6.1) to (4.6.4) (for all $m,n,p,q\geq 0$, all $s$ in $J_{m,n}$, all $t$ in $J_{n,p}$, all $u$ in $J_{p,q}$, all $i$ in $[n]$; $O_p$ is another notation for $(\;)$ so that (4.6.4) says that $J_{0,p}$ is reduced to $O_p$).

One assumes that $\pi_{i,p}\neq\pi_{j,p}$ if $i\neq j$.

The theory is *ideal* if for all $u$ in $J_{1,p}$ if $u$ is not distinguished (i.e. $u\neq\pi_{i,p}$ for all $i\in[p]$) then for all $t$ in $J_{p,n}$, $u\cdot t$ is not distinguished. An object $u$ in $J_{n,p}$ is *ideal* if for all $i\in[n]$, $\pi_i\cdot u$ is not distinguished.

The theories 'of trees' $T$, $R$ and $M$ are ideal and an $n$-tree $(u_1,\ldots,u_n)$ is ideal if and only if none of $u_1,\ldots,u_n$ is a variable.

An ideal theory $J$ is *scalar iterative* if for every ideal $u$ in $J_{1,1+p}$ the equation (4.6.6) has a unique solution in $J_{1,p}$ (it will be denoted by $u^{\dagger}$). It is *(vector) iterative* if the same holds for every ideal $u$ in $J_{n,n+p}$ ($n\geq 1$, $p\geq 0$). The solution $u^{\dagger}$ is then in $J_{n,p}$.

**Remarks 4.7.1.** (1) In the case of trees, the condition "$u$ is ideal" is stronger than the condition "$\pi_i\cdot u\neq\pi_j$ for all $i,j$ in $[n]$" that was used in Theorem 4.3.1 and in (4.6.6) since the latter allows $u_i$ in $\{v_{n+1},v_{n+2},\ldots,v_{n+p}\}$.

(2) The results we mentioned in Remark 4.3.2 are actually proved in [7] for arbitrary ideal theories.

The method we used in Section 4.4 to solve systems equation by equation is applicable in any scalar iterative theory. The unicity of the solution of (4.6.6) can also be proved from the unicity in the scalar case. Hence

**Proposition 4.7.2** ([8]). *A theory is iterative if and only if it is scalar iterative.*

### 4.7.3. *Iterative theory expressions*

Let $J$ be an iterative theory, $F$ a ranked alphabet, $\nu$ a mapping: $F \to J$ such that $\nu(f)$ is an ideal element of $J_{1,k}$ for all $f$ in $F_k$. We shall say that *f denotes* $\nu(f)$.

We shall now define *iterative theory expressions over F*, each of them having a *type* $(n, p)$, in such a way that $e$ of type $(n, p)$ denotes an element $e_{J,\nu}$ of $J_{n,p}$ (which may be undefined, see below).

The set $E_F(n, p)$ of (iterative theory) expressions over $F$ of type $(n, p)$ is defined as follows:

- $O_p$ belongs to $E_F(0, p)$,
- $\pi_{i,p}$ belongs to $E_F(1, p)$,
- $(e_1, \ldots, e_n)$ belongs to $E_F(n, p)$ if $e_1, \ldots, e_n$ belong to $E_F(1, p)$,
- $e \cdot e'$ belongs to $E_F(n, q)$ if $e$ belongs to $E_F(n, p)$ and $e'$ belongs to $E_F(p, q)$,
- $e^\dagger$ belongs to $E_F(n, p)$ if $e$ belongs to $E_F(n, n+p)$,
- $f$ belongs to $E_F(1, k)$ if $f \in F_k$.

The formal definition of $e_{J,\nu}$ can be given by induction on the structure of $e$ in an obvious way, with the requirement that $(e^\dagger)_{J,\nu}$ is defined only if $e_{J,\nu}$ is defined and ideal.

Let us detail a few rules:

- $(e \cdot e')_{J,\nu} = e_{J,\nu} \cdot e'_{J,\nu}$ if $e_{J,\nu}$ and $e'_{J,\nu}$ are both defined and undefined otherwise,
- $(e^\dagger)_{J,\nu} = (e_{J,\nu})^\dagger$ is defined *and is ideal* and undefined otherwise.

One could also declare that $(e^\dagger)_{J,\nu}$ is defined if and only if $e_{J,\nu}$ is defined and the equation $u = e_{J,\nu} \cdot (u, I_p)$ has a unique solution in $J$, but we shall not use this alternative definition.

Note also that we do not distinguish constant and operation symbols (i.e. $O_p$, $\pi_{i,p}$, $( , \ldots , )$, $^\dagger$, $\cdot$ ) from what they denote in $J$.

Let us finally mention that iterative theory expressions have been used by Bloom and Elgot [6] to define and construct free iterative theories.

## 4.8. *The free iterative theory generated by F*

The definition of a *homomorphism* $\Psi : J \to J'$ of algebraic theories is obvious: $\Psi$ must map $J_{n,p}$ into $J'_{n,p}$ and must preserve operations, distinguished elements etc.... We say that $\Psi$ is *ideal* if $J$ and $J'$ are ideal and $\Psi$ maps an ideal object onto an ideal one.

Let $\nu$ be a family of mappings: $F \to J$ as in Section 4.7.3. It is easy to see that $\nu$ extends uniquely into a family of mappings $\Psi_{1,p} : M(F, V_p) \to J_{1,r}$ such that

$$\Psi_{1,p}(v_i) = \pi_{i,p},$$

$$\Psi_{1,p}(f(t_1, \ldots, t_k)) = \nu(f) \cdot (\Psi_{1,p}(t_1), \ldots, \Psi_{1,p}(t_k))$$

and by tupling into mappings $\Psi_{n,p} : M(F, V_p)^n \to J_{n,p}$ defining a unique homomorphism: $M_F \to J$.

This shows that $M_F$ is the *free algebraic theory* generated by $F$.

If $J$ is iterative and since $\nu(f)$ is ideal for all $f$ in $F$, it can be shown that $\Psi$ extends uniquely into a homomorphism: $R_F \to J$. Hence

**Theorem 4.8.1** ([34, 40]). *$R_F$ is the free iterative theory generated by $F$.*

Since an ideal homomorphism of iterative theories $\Psi$ maps $u^\dagger$ onto $(\Psi(u))^\dagger$ we *must* define $\Psi_{n,p}(u^\dagger)$ as $(\Psi_{n,n+p}(u))^\dagger$ for $u$ in $M_{n,n+p}$. And this (together with tupling) defines $\Psi_{n,p}$ for all element of $R_{n,p}$. But we must show that $\Psi_{n,p}$ is well defined, i.e. that if $u^\dagger = w^\dagger$ then $(\Psi_{n,n+p}(u))^\dagger = (\Psi_{n,n+p}(w))^\dagger$.

This is the crux of the proof of Theorem 4.8.1 (see [34, 40]).

We shall denote by $e_F$ the value of an expression $e$ in $R_F$ if $f$ denotes $\bar{f}$ for all $f$ in $F$.

**Corollary 4.8.2.** *For every iterative theory $J$, every ideal mapping $\nu : F \to J$, every expression $e$:*

(1) *$e_{J,\nu}$ is defined if and only if $e_F$ is defined,*

(2) *if $e_F$ is defined then $e_{J,\nu} = \Psi(e_F)$ where $\Psi : R_F \to J$ is the unique homomorphism extending $\nu$.*

**Proof.** (1) and (2) can be proved simultaneously by induction on the structure of $e$. Note in particular that

$(e^\dagger)_{J,\nu}$ is defined if and only if

$e_{J,\nu}$ is defined and ideal if and only if

$e_F$ is defined and ideal (since $e_{J,\nu} = \Psi(e_F)$ by induction and $\Psi$ is ideal) if and only if

$(e^\dagger)_F$ is defined. □

Two expressions are *equivalent* if in all iteratitive theory, either they denote the same thing or they are both undefined.

**Corollary 4.8.3.** *Two expressions $e$ and $e'$ of same type are equivalent if and only if $e_F = e'_F$.*

Let us also remark that Cousineau's rational expressions can be defined as the subset $\mathrm{RSE}_F$ of *restricted scalar expressions over $F$*, inductively defined as follows:

- $\pi_{i,p}$ belongs to $\mathrm{RSE}_F(p)$,
- $e^\dagger$ belongs to $\mathrm{RSE}_F(p)$ if $e$ belongs to $\mathrm{RSE}_F(p+1)$,
- $f \cdot (e_1, \ldots, e_k)$ belongs to $\mathrm{RSE}_F(p)$ if $f \in F_k$ and $e_1, \ldots, e_k$ belong to $\mathrm{RSE}_F(p)$ (for $k = 0$ we have $f \cdot 0_p$ in $\mathrm{RSE}_F(p)$).

**Theorem 4.8.4.** *For every expression $e$ over $F$ of type $(n, p)$, one can find restricted scalar expressions $e'_1, \ldots, e'_n$ over $F$ and of type $p$, such that $(e'_1, \ldots, e'_n)$ is equivalent to $e$.*

**Proof.** If $e_F$ is undefined there is nothing to do. Otherwise $e_F = (t_1, \ldots, t_n)$. Theorem 4.5.7 shows that one can find rational expressions $e''_1, \ldots, e''_n$ such that $\mathbf{Val}(e''_i) = t_i$ for all $i$. The translation of $e''_i$ into $e'_i$ in $\mathrm{RSE}_F(p)$ such that $e'_{iF} = \mathbf{Val}(e''_i)$ is obvious so that $e_F = (e'_1, \ldots, e'_n)_F$.

Corollary 4.8.3 shows that $e$ and $(e'_1, \ldots, e'_n)$ are equivalent. □

The above connection between Cousineau's theorem and iterative theories is a new result.

Elgot proves in [32, part 2 of the main theorem] and [33, Theorem 4.1] a similar result with a slightly larger class of scalar expressions.

This class, let us denote it by $\mathrm{R'SE}_F$, is defined by the same rules as $\mathrm{RSE}_F$ together with:

$e \cdot e'$ belongs to $\mathrm{R'SE}_F(p)$ if $e$ belongs to $\mathrm{R'SE}_F(1)$ and $e'$ belongs to $\mathrm{R'SE}_F(p)$.

This corresponds to the *composition* of program schemes denoted by $e$ and $e'$, whereas $f \cdot (e_1, \ldots, e_k)$ corresponds to *alternation* [33].

Introducing the composition is meaningful if one wants to write program schemes in a structured way but not necessary as shown by Theorem 4.8.4.

**Remark 4.8.5.** A homomorphism of free iterative theories: $R_F \to R_G$ is nothing else than the extension by tupling of a second-order substitution which is nonerasing and regular, i.e. such that $\nu(f) \in R(G, V_k) - V_k$ for all $f$ in $F_k$, all $k \geq 0$.

### *4.9. First-order unification of infinite trees*

Let $t, t'$ be two trees in $M(F, V_k)$. A (*first-order*) *unifier* of $t$ and $t'$ is a (first-order) substitution $\sigma : M(F, V_k) \to M(G)$ for some $G \supseteq F$ such that $\sigma(t) = \sigma(t')$. We shall denote by $\mathbf{Unif}_{k,G}(t, t')$ the set of all such substitutions.

Determining $\mathbf{Unif}_{k,G}(t, t')$ corresponds to finding all solutions in $M(G)$ of the equation $t = t'$ the unknowns of which are $v_1, v_2, \ldots, v_k$.

Proposition 3.4.2 shows that if $\sigma : V_k \to M(F, X_l)$ is a unifier of $t$ and $t'$ and $\tau$ is any substitution: $X_l \to M(G)$ then $\tau \cdot \sigma$ (also denoted by $\tau\sigma$) is also a unifier of $t$ and $t'$. We shall say that $\tau\sigma$ *is deduced from* $\sigma$ or that $\sigma$ *is more general than* $\tau\sigma$.

**Theorem 4.9.1.** *If $t$ and $t'$ in $M(F, V_k)$ are unifiable they have a most general unifier $\sigma : V_k \to M(F, X_l)$. One can find one with $l \leq k$.*

This means that, for every ranked alphabet $G \supseteq F$,

$$\mathbf{Unif}_{k,G}(t, t') = \{\tau\sigma / \tau : X_l \to M(G)\}.$$

Let us consider two most general unifiers $\sigma: V_k \to M^\infty(G, X_l)$ and $\sigma': V_k \to M^\infty(G, Y_{l'})$ (where $X_l = \{x_1, \dots, x_l\}$ and $Y_{l'} = \{y_1, \dots, y_{l'}\}$). There exists $\theta: X_l \to M(G, Y_{l'})$ and $\theta': Y_{l'} \to M^\infty(G, X_l)$ such that $\sigma' = \theta\sigma$ and $\sigma = \theta'\sigma'$ hence $\sigma = \theta'\theta\sigma$ and $\theta'\theta$ is the identity on the set $\mathbf{Var}(\sigma) = \bigcup\{\mathbf{Var}(\sigma(v_i)) \mid 1 \leq i \leq k\}$ by Proposition 3.4.1 part (1). Similarly $\theta\theta'$ is the identity on $\mathbf{Var}(\sigma')$.

This shows that if $\sigma$ and $\sigma'$ are such that $\mathbf{Var}(\sigma) = X_l$ and $\mathbf{Var}(\sigma') = Y_{l'}$, i.e. such that all variables in $X_l$ and $Y_{l'}$ are useful, then $X_l$ and $Y_{l'}$ are in bijection by $\theta$ and, in particular $l = l'$. We shall say that $\theta$ is an $X_l$-renaming.

The integer $\mathbf{Card}(\mathbf{Var}(\sigma))$ is the minimal $l$ such that there exists a most general unifier of $t$ and $t'$ of the form: $V_k \to M(G, X_l)$. Note that this integer does not depend only on $t$ and $t'$ but also on $k$. We call it the *rank* of $(t, t')$ when $V_k = \mathbf{Var}(t) \cup \mathbf{Var}(t')$. Clearly,

$$\mathbf{Rank}(t, t') = \mathbf{Card}(\bigcup\{\mathbf{Var}(\sigma(v)) \mid v \in \mathbf{Var}(t) \cup \mathbf{Var}(t')\})$$

where $\sigma$ is any most general unifier of $t$ and $t'$. This is the number of independent parameters upon which the general solution in finite trees of the equation $t = t'$ depends. By Theorem 4.9.1, $\mathbf{Rank}(t, t') \leq \mathbf{Card}(\mathbf{Var}(t) \cup \mathbf{Var}(t'))$.

The proof of Theorem 4.9.1 also shows that the function symbols occurring in $\sigma \restriction (\mathbf{Var}(t) \cup \mathbf{Var}(t'))$ are all in $F$ so that the chosen alphabet $G$ is irrelevant provided $F \subseteq G$.

Theorem 4.9.1 has a special interest in mechanical theorem proving (Robinson [61]). A linear algorithm has been given by Paterson and Wegman [60] to construct the most general unifier of two terms or show that it does not exist.

Our intention is to extend this theorem to infinite trees. The results we shall present are essentially due to Huet [48].

Let $t, t' \in M^\infty(F, V_k)$. A *unifier* of $t$ and $t'$ is a substitution $\sigma: V_k \to M^\infty(G)$ such that $\sigma(t) = \sigma(t')$. We shall denote by $\mathbf{Unif}^\infty_{k,G}(t, t')$ the set of such unifiers. (We shall often omit the mention of $k$ and $G$.)

Such a substitution $\sigma$ is *finite* (resp. *regular*) if $\sigma(v)$ is finite (resp. regular) for all $v$ in $V_k$.

As in the finite case we say that a unifier $\sigma$ is *more general* than a unifier $\tau\sigma$ and we define most general unifiers in an obvious way.

**Theorem 4.9.2.** *Let $t$ and $t'$ belong to $M^\infty(F, V_k)$.*

(1) *If $t$ and $t'$ are unifiable, they have a most general unifier $\tau: V_k \to M^\infty(F, X_l)$, and $l \leq k$. It is unique up to an $X_l$-renaming when $l$ is minimal.*

(2) *If $t$ and $t'$ are regular and unifiable then their most general unifier is regular. It can be effectively computed.*

The proof of Theorem 4.9.2 requires a number of technical definitions.

We shall consider sets $\mathscr{C}$ of pairs of trees in $M^\infty(F, V_k)$ and define $\mathbf{Unif}^\infty_{k,G}(\mathscr{C})$ as the set of substitutions: $V_k \to M^\infty(G)$ which unify all pairs $(t, t')$ in $\mathscr{C}$, and we define most general unifiers in an obvious way.

Let $\mathcal{T}$ be the set of all subtrees of the components of all pairs in $\mathscr{C}$. An equivalence relation $\sim$ on $\mathcal{T}$ is *simplifiable* if, for all $t$, $t'$ in $\mathcal{T}$ such that

$$t = f(t_1, \ldots, t_k),$$

$$t' = f(t'_1, \ldots, t'_k),$$

$$t \sim t',$$

we have $t_i \sim t'_i$ for all $i = 1, \ldots, k$.

It is *coherent* if we do not have $f(t_1, \ldots, t_k) \sim g(t'_1, \ldots, t'_l)$, $f, g \in F$ and $f \neq g$ for any $t_1, \ldots, t_k, t'_1, \ldots, t'_l$.

We shall denote by $\sim_{\mathscr{C}}$ the least simplifiable equivalence relation on $\mathcal{T}$ which contains $\mathscr{C}$. It does exist and standard arguments yield the following lemma:

**Lemma 4.9.3.** $\mathbf{Unif}^{\infty}(\mathscr{C}) = \mathbf{Unif}^{\infty}(\sim_{\mathscr{C}})$.

Let us assume that $\sim_{\mathscr{C}}$ is coherent and let $S$ be the generalized regular system of equations defined as follows (we shall denote $\sim_{\mathscr{C}}$ by $\sim$).

We let $X$ be the set of variables $v$ in $V$ such that.

(i) $v \sim t$ for some $t$ such that $\mathbf{First}(t) \in F$,

(ii) (i) does not hold and $v = v_i$, $v_i \sim v_j$ for some $j > i$.

We let $W$ be $V_k - X$ and $S$ be the system of equations $\langle v = u\,;\, v \in X\rangle$ such that if $v$ satisfies (i) then $u = t$ and if $v$ satisfies (ii) then $u = v_j$ where $j$ is the largest index such that $v \sim v_j$.

**Lemma 4.9.4.** *$S$ is a generalized regular Greibach system over $F \cup W$ with set of unknowns $X$.*

**Proof.** We have to show that any $v_j$ occurring in an equation $v = v_j$ of $S$ belongs to $W$. If this was not true, then either $v_j \sim t$ for some $t$ such that $\mathbf{First}(t) \in F$ and $v \sim t$, and $v$ would have to satisfy (i) which is not the case, or $v_j \sim v_{j'}$ for some $j' > j$ and then, $v \sim v_{j'}$ and $j$ would not be maximal as required in the definition of $S$. □

Without loss of generality we can assume that $X = \{v_1, v_2, \ldots, v_l\}$ and $W = \{v_{l+1}, \ldots, v_k\}$. We let $(t_1, \ldots, t_l)$ denote the unique solution of $S$ in $M^{\infty}(F, W)$ and $\tau : X \cup W \to M^{\infty}(F, W)$ the substitution such that $\tau(v_i) = t_i$ for $v_i \in X$ and $\tau(v_i) = v_i$ for $v_i \in W$.

**Proposition 4.9.5.** (1) $\mathbf{Unif}^{\infty}(\mathscr{C}) \neq \emptyset$ *if and only if* $\sim_{\mathscr{C}}$ *is coherent.*

(2) *If* $\mathbf{Unif}^{\infty}(\mathscr{C}) \neq \emptyset$ *then $\tau$ is the most general unifier of* $\mathscr{C}$.

**Proof.** If $\sim_{\mathscr{C}}$ is not coherent then $\mathbf{Unif}^{\infty}(\sim_{\mathscr{C}}) = \emptyset$ and $\mathbf{Unif}^{\infty}(\mathscr{C}) = \emptyset$ by Lemma 4.9.3. Otherwise, let $S$ be defined as above. By considering $S$ as a set of pairs in $M^{\infty}(F, V)$, we can consider $\mathbf{Unif}^{\infty}(S)$. Let us show that it coincides with $\mathbf{Unif}^{\infty}(\mathscr{C})$.

Since $S \subseteq \sim_{\mathscr{C}}$ we have $\mathbf{Unif}^{\infty}(S) \supseteq \mathbf{Unif}^{\infty}(\mathscr{C})$ (with help of Lemma 4.9.3).

Let now $\sigma : V_k \to M^{\infty}(G)$ be a unifier of $S$. Let us show that $\sigma \in \mathbf{Unif}^{\infty}(\sim_{\mathscr{C}})$ and Lemma 4.9.3 will give us the desired result.

We show that $\delta(\sigma(t), \sigma(t')) = \infty$ for all $t, t'$ such that $t \sim t'$ (we denote $\sim_{\mathscr{C}}$ by $\sim$). We do this by contradiction, letting $n$ be the minimal $n < \infty$ (if any) such that $\delta(\sigma(t), \sigma(t')) = n$ for some $t, t'$ such that $t \sim t'$.

*Case 1*: $\mathbf{First}(t)$ and $\mathbf{First}(t') \in F$. We necessarily have $\mathbf{First}(t) = \mathbf{First}(t')$ (since otherwise $\sim$ is not coherent), hence $t = f(t_1, \ldots, t_m)$, $t' = f(t'_1, \ldots, t'_m)$ and there exists $i$ in $[m]$ such that $\delta(\sigma(t_i), \sigma(t'_i)) = n - 1$. Hence $t_i \sim t'_i$ and this contradicts the minimality of $n$.

*Case 2*: $\mathbf{First}(t) \in F$ and $t' = v_i$. By definition of $S$, $v_i$ is in $X$ and the corresponding equation of $S$ is of the form $v_i = u$ for some $u$ in $M^{\infty}(F, V)$ with $\mathbf{First}(u) \in F$. Since $t \sim v_i$ and $v_i \sim u$ we have $t \sim u$. Since $\sigma$ unifies $S$, $\sigma(v_i) = \sigma(u)$ hence $\delta(\sigma(t), \sigma(t')) = \delta(\sigma(t), \sigma(u))$. We are back to the first case, which cannot happen as we have just seen.

*Case 3*: $t = v_i$, $t' = v_j$ and we can assume that $i < j$. By definition of $S$, we cannot have $v_i$ and $v_j$ both in $W$. Hence $v_i \in X$. Let $v_i = u$ be the corresponding equation of $S$. Note that $u \sim v_j$.

If $\mathbf{First}(u) \in F$ then $v_j$ must be in $X$; let $v_j = u'$ be the corresponding equation of $S$; we have $\mathbf{First}(u')$ in $F$ and $u \sim u'$. Since $\sigma$ unifies $S$, $\sigma(v_i) = \sigma(u)$ and $\sigma(v_j) = \sigma(u')$ hence $\delta(\sigma(t), \sigma(t')) = \delta(\sigma(u), \sigma(u'))$ and we are back to the Case 1. Contradiction.

Hence $u \in W$ and necessarily, $u = v_j$. Since $\sigma$ unifies $S$, $\sigma(v_i) = \sigma(v_j)$ hence $\sigma(t) = \sigma(t')$ and we cannot have $\delta(\sigma(t), \sigma(t')) = n < \infty$.

Hence we have shown that

$$\mathbf{Unif}^{\infty}(S) = \mathbf{Unif}^{\infty}(\mathscr{C}).$$

It is clear that $\mathbf{Unif}^{\infty}(S)$ is not empty: it contains at least $\tau$.

In order to achieve the proof of (2) we need only show that

$$\mathbf{Unif}^{\infty}_{k,G}(S) = \{\sigma\tau / \sigma : W \to M^{\infty}(G)\}$$

for all ranked alphabet $G$ including $F$.

For every equation $v_i = u_i$ of $S$ $(i = 1, \ldots, l)$ we have

$$\begin{aligned}\sigma\tau(v_i) &= \sigma(\tau(v_i)) \\ &= \sigma(\tau(u_i)) \quad \text{(by definition of } \tau) \\ &= \sigma\tau(u_i).\end{aligned}$$

Conversely, let $\sigma' : X \cup W \to M^{\infty}(G)$ such that $\sigma'(v_i) = \sigma'(u_i)$ for all $i = 1, \ldots, l$. Let $\sigma$ be the restriction of $\sigma'$ to $W$. Just to simplify the proof and without loss of generality we can assume that $G \cap X = \emptyset$. We have

$$\sigma'(v_i) = \sigma(u_i)[\sigma'(v_1)/v_1, \ldots, \sigma'(v_l)/v_l] \quad (i = 1, \ldots, l).$$

We also have

$$\sigma(t_i) = \sigma(u_i[t_1/v_1, \ldots, t_l/v_l])$$
$$= \sigma(u_i)[\sigma(t_1)/v_1, \ldots, \sigma(t_l)/v_l].$$

Since the system $\langle v_i = \sigma(u_i);\ 1 \leq i \leq l\rangle$ clearly has a unique solution:

$$\sigma'(v_i) = \sigma(t_i) = \sigma\tau(v_i) \quad \text{for } i = 1, \ldots, l.$$

Also

$$\sigma'(v_i) = \sigma(v_i) = \sigma\tau(v_i) \quad \text{for } i = l+1, \ldots, k$$

since $\tau$ is the identity on $W$.

Hence $\sigma' = \sigma\tau$ for some substitution $\sigma : W \to M^\infty(G)$ as wanted. □

**Proof of Theorem 4.9.2.** (1) is consequence of Proposition 4.9.5 with $\mathscr{C} = \{(t, t')\}$.

All the remarks we made in the finite case on the minimality of $l$ also apply since they were only depending on Proposition 3.4.1 part (1) which holds for infinite trees.

The *rank* of a pair $(t, t')$ can be defined as well and $\textbf{Rank}(t, t') \leq \textbf{Card}(\textbf{Var}(t) \cup \textbf{Var}(t'))$.

(2) If $\mathscr{C}$ is a set of pairs of regular trees, then $\mathscr{T}$ is a set of regular trees and $S$ is an extended regular system. Its solution consists of regular trees hence $\tau$ is regular.

Furthermore if $t$ and $t'$ are two given regular trees, the set $\mathscr{T}$ associated with $\mathscr{C} = \{(t, t')\}$ is finite ($\mathscr{T} = \textbf{Subtree}(t) \cup \textbf{Subtree}(t')$) and can be effectively constructed. The equivalence $\sim_\mathscr{C}$ can be computed and tested for coherence. If it is coherent then $S$ and $\tau$ can be effectively determined. More details concerning this algorithm can be found in Huet [48]. □

The above technique will be extended to the problem of determining whether $\textbf{Unif}(t, t') = \emptyset$ for $t, t'$ in $M(F, V)$ and more generally in $R(F, V)$.

It is clear that $\textbf{Unif}(t, t') = \emptyset$ if $\textbf{Unif}^\infty(t, t') = \emptyset$. If $\textbf{Unif}^\infty(t, t') \neq \emptyset$, deciding whether $\textbf{Unif}(t, t') = \emptyset$ amounts to deciding whether the most general unifier $\tau$ of $t$ and $t'$ is finite.

To do so, we shall use the equivalence $\sim_\mathscr{C}$ constructed in the proof of Proposition 4.9.5.

We define a binary relation $\Rightarrow$ on $V$ by letting $v_i \Rightarrow v_j$ if and only if $v_i \sim_\mathscr{C} t$ for some $t$ in $M^\infty(F, V)$ such that $\textbf{First}(t) \in F$ and $v_j$ has at least one occurrence in $t$.

We say that $\sim_\mathscr{C}$ is *acyclic* [48] if $v \overset{+}{\Rightarrow} v$ for no $v$ in $V$.

**Proposition 4.9.6.** *Let $\mathscr{C}$ be a finite set of pairs of regular trees.* $\textbf{Unif}(\mathscr{C}) \neq \emptyset$ *if and only if $\sim_\mathscr{C}$ is coherent and acyclic. This property is decidable.*

**Proof.** If $\sim_\mathscr{C}$ is not acyclic then $\textbf{Unif}(\mathscr{C}) = \emptyset$ (if $v \overset{+}{\Rightarrow} v$ and $\sigma \in \textbf{Unif}(\mathscr{C})$, $\sigma(v)$ satisfies an equation of the form $x = t$ where $t \neq x$ and $x$ has an occurrence in $t$; this is impossible if $\sigma(v)$ is finite).

If $\mathbf{Unif}(\mathscr{C}) \neq \emptyset$ then $\mathbf{Unif}^{\infty}(\mathscr{C}) \neq \emptyset$ hence $\sim_{\mathscr{C}}$ is coherent (by Proposition 4.9.5). Note also that $\mathbf{Unif}(\mathscr{C}) = \mathbf{Unif}(\sim_{\mathscr{C}})$, hence $\sim_{\mathscr{C}}$ is acyclic by the above remark.

Conversely, if $\sim_{\mathscr{C}}$ is coherent and acyclic the system $S$ is such that its unique solution is finite. Hence the most general unifier of $\mathscr{C}$ is finite and $\mathbf{Unif}(\mathscr{C}) \neq \emptyset$.

This property is decidable in nearly linear time. See Huet [48] for algorithms. □

**Example 4.9.7.** Let us consider the equation $f(v_1, f(v_3, v_2)) = f(h(v_2, v_1), f(v_4, g(v_3, v_4)))$. The construction of $\sim$ gives us

$$v_1 \sim h(v_2, v_1), \qquad v_3 \sim v_4, \qquad v_2 \sim g(v_3, v_4)$$

whence $X = \{v_1, v_2, v_3\}$ and $W = \{v_4\}$ and $S$ is the following system:

$$\langle v_1 = h(v_2, v_1), v_2 = g(v_3, v_4), v_3 = v_4 \rangle.$$

Its solution is the triple $(t_1, t_2, t_3)$ of trees in $R(F \cup \{v_4\})$ such that

$$t_1 = h(g(v_4, v_4), h(g(v_4, v_4), h(g(v_4, v_4), \ldots))),$$

$$t_2 = g(v_4, v_4),$$

$$t_3 = v_4.$$

Hence the most general unifier of the given pair of trees, i.e. the general solution of the given equation is the substitution $\tau : V_4 \to R(F, \{v_4\})$ such that $\tau(v_i) = t_i$ for $i = 1, 2, 3$ and $\tau(v_4) = v_4$. Its rank is 1.

We shall apply this technique to prove the characterization of *all* solutions of a nonproper regular system that we gave in Remark 4.3.2.

Let $\mathscr{C} = \langle v_i = u_i; 1 \leq i \leq n \rangle$ be a generalized regular system where $u_i \in M^{\infty}(F, V_n)$ for all $i$. Let $\mathscr{T} = V_n \cup \{\mathbf{Subtree}(u_i) \mid 1 \leq i \leq n\}$. Let us associate with $\mathscr{C}$ a binary relation $\to_{\mathscr{C}}$ on $M^{\infty}(F, V_n)$ defined as follows:

$t \to_{\mathscr{C}} t'$ if and only if there exists some $w$ in $M^{\infty}(F, V_{n+1})$ with exactly one occurrence of the auxiliary variable $v_{n+1}$ and such that, for some $i$ in $[n]$,

$$t = w[v_i/v_{n+1}], \qquad t' = w[u_i/v_{n+1}].$$

**Lemma 4.9.8.** *For all $t, t'$ in $\mathscr{T}$, if $t \sim t'$ then $t \xrightarrow{*}_{\mathscr{C}} u$ and $t' \xrightarrow{*}_{\mathscr{C}} u$ for some $u$ in $M^{\infty}(F, V_n)$.*

**Proof.** The binary relation $\approx$ on $M^{\infty}(F, V_n)$ such that:

$$s \approx s' \text{ if and only if } s \xrightarrow{*}_{\mathscr{C}} u,\ s' \xrightarrow{*}_{\mathscr{C}} u \text{ for some } u \text{ in } M^{\infty}(F, V_n)$$

is reflexive, symmetric and simplifiable (easy to show). Although it is defined on infinite trees, the methods of Rosen [62], O'Donnell [57] or Huet [49] allow to show that $\xrightarrow{*}_{\mathscr{C}}$ is confluent (i.e. has the Church–Rosser property) hence that the relation $\approx$ is transitive.

Since $\approx$ contains $\mathscr{C}$, it also contains $\sim$. □

By using Lemma 4.9.8 one can easily prove the following facts:

**Fact 1.** *$\sim$ is coherent.*

**Fact 2.** *For every $v$ in $V_n$, the following conditions are equivalent*:
(i) *$v \xrightarrow{*}_{\mathscr{C}} v'$ for some singular variable $v'$,*
(ii) *for all $t$ in $\mathscr{T}$, if $v \sim t$ then $t \in V$.*

**Fact 3.** *For any two variables $v$ and $v'$ satisfying the conditions of Fact 2, $v \sim v'$ if and only if $v \xrightarrow{*}_{\mathscr{C}} v''$ and $v' \xrightarrow{*}_{\mathscr{C}} v''$ for some singular variable $v''$.*

Let us finally modify slightly the definition of $S$ given in Lemma 4.9.4:
- $W = \{v_j / v_j$ is singular and for all singular $v_{j'}$ if $v_{j'} \sim v_j$ then $j' < j\}$
- $X = V_n - W$,
- $S' = \langle v = u ; v \in X \rangle$ where, for each $v$ in $X$,
  (i) either $u$ is some element of $M^\infty(F, V_n)$ such that $v \sim u$ and **First**$(u) \in F$,
  (ii) or $u \in W$ with $v \sim u$.

This system obviously satisfies the Greibach condition and it is easy to show that its unique solution defines the most general unifier of $\mathscr{C}$ as in Proposition 4.9.5. Whence the characterization of all solutions of $\mathscr{C}$ that we mentioned in Remark 4.3.2 (Bloom *et al.* [7]).

**Example 4.9.9.** Let $\mathscr{C}$ be the following system:

$$v_1 = h(v_2, v_1), \qquad v_2 = g(v_3, v_4), \qquad v_3 = v_4, \qquad v_4 = v_3.$$

The singular unknowns are $v_3$ and $v_4$. The system $S'$ is exactly the system $S$ of Example 4.9.7.

**Remark 4.9.10.** For regular trees $t$ and $t'$ in $M^\infty(F)$, **Unif**$(t, t') \neq \emptyset$ if and only if $t = t'$. A corollary of Proposition 4.9.6 is the decidability of the equality problem for regular trees.

*4.10. Least solutions of regular systems in $M_\Omega^\infty(F)$*

Here we shall consider *possibly nonproper* generalized and extended regular systems.

Since $\Omega$ is a constant, we can define the sets $R(F \cup \{\Omega\})$ and $R(F \cup \{\Omega\}, V)$ of regular trees. We shall denote them by $R_\Omega(F)$ and $R_\Omega(F, V)$ to emphasize the special role of $\Omega$.

Let $S = \langle x_1 = u_1, \ldots, x_n = u_n \rangle$ be a generalized regular system with $u_i \in M^\infty(F, X_n)$ for $i = 1, \ldots, n$. We do not require that **First**$(u_i) \in F$.

**Theorem 4.10.1.** *$S$ has a least solution in $M_\Omega^\infty(F)$. If $S$ is extended regular, all components of its least solution are regular.*

**Proof.** The mapping $|S|: M_\Omega^\infty(F)^n \to M_\Omega^\infty(F)^n$ is monotone and $\omega$-continuous. Hence $|S|$ has a least fix-point, i.e. $S$ has a least solution. Furthermore, this least solution can be defined as the least upper bound of $|S|^n(\Omega, \ldots, \Omega)$ for $n \geq 0$.

Since $R_\Omega(F)$ is *not* $\omega$-complete, the second assertion does not follow immediately.

Let $S = \langle x_1 = u_1, \ldots, x_n = u_n \rangle$ be an extended regular, possibly nonproper system and let $(t_1, \ldots, t_n)$ be its least solution in $M_\Omega^\infty(F)$. Let $S'$ be the proper system $\langle x_1 = u'_1, \ldots, x_n = u'_n \rangle$ such that $u'_i = \Omega$ if $t_i = \Omega$ and $u'_i = u_i$ if $t_i \neq \Omega$.

It can be checked that $S$ and $S'$ have the same least solution (for all $i \geq 0$, $|S|^i(\Omega, \ldots, \Omega) = |S'|^i(\Omega, \ldots, \Omega)$; this can be shown by induction on $i$).

It can be shown that $S'$ is proper: if $x_i$ is singular in $S$ then $t_i = \Omega$ hence $x_i$ is no more singular in $S'$. In fact $S'$ has no singular unknown hence it has a unique solution. Hence Theorem 4.3.1 is applicable and shows that $(t'_1, \ldots, t'_n)$, the least and unique solution of $S'$ belongs to $R(F \cup \{\Omega\})^n$. (Note that in our application of Theorem 4.3.1, $\Omega$ is used as an ordinary constant). □

By *solving* a system of equations we shall mean here determining its least solution in $M_\Omega^\infty(F)$.

Generalized regular systems can be solved equation by equation, exactly as in Section 4.4, with $M_\Omega^\infty(F, \{x_2, \ldots, x_n\})$ in place of $M^\infty(F, \{x_2, \ldots, x_n\})$ and least solutions in place of unique ones. The validity of the method, i.e. the analog of Claim 4.4.1, is a direct consequence of Lemma 2.3.2.

Cousineau's rational expressions can also be used (actually they are defined in [29] so as to define trees in $R_\Omega(F)$ and not only in $R(F)$) with the following changes.

One defines **Star**$(t)$ as the least tree in $M_\Omega^\infty(F, V_{k-1})$ (see Definition 4.5.1) such that **Star**$(t) = t[$**Star**$(t)/v_1, v_1/v_2, \ldots, v_{k-1}/v_k]$, so that **Star**$(v_1) = \Omega$. Lemma 4.5.2 still holds but the proof is a bit more technical. (Having lost the unicity property defining **Star**$(t)$, one uses the characterization of **Star**$(t)$ as the least upper bound of the sequence of iterates of $t$, i.e. **Star**$(t) =$ **Sup**$_n(u_n)$ where $u_0 = \Omega$, $u_{n+1} = t[u_n, v_1, \ldots, v_{k-1}]$. See [29] for a detailed proof.)

Rational expressions are now elements of $M(F \cup \{*, \Omega\}, V)$, and each of them has a value in $R_\Omega(F, V)$. Note that if $e$ has a value in the sense of Definition 4.5.3, it has the same value in the new sense. The mappings **Comp** and **Shift** extend immediately (we consider $\Omega$ as a constant) and the analogs of Claims 4.5.5 and 4.5.6 also hold (the proofs are the same). Hence, Theorem 4.5.7 also extends to $R_\Omega(F, V)$ and applies to possibly nonproper extended regular systems and to rational expressions with $\Omega$.

Finally all what we said concerning iterative theories has a counterpart in *rational theories*.

A rational theory (defined in ADJ [69]) is an algebraic theory $A$ where each set $A_{n,p}$ is partially ordered and has a least element. The requirement of existence and unicity of a solution of (4.6.6) is replaced by the requirement of existence of a least solution, without any limitation to ideal equations. The reader is referred to [69] for more details on rational theories.

Tiuryn has defined similar objects named regular algebras [65] and investigated their relations with iterative and rational theories in [66, 67].

### 4.11. *Branch languages and occurrence languages of regular trees*

The relation between regular trees and regular languages is very natural as shown by the following theorem:

**Theorem 4.11.1.** (1) *A tree $t$ in $M^\infty(F)$ is regular if and only if* **Occ**$(f, t)$ *is regular for all $f$ in $F$ and empty for all but finitely many of them, if and only if* **PBrch**$(t)$ *is regular.*

(2) *A tree $t$ in $M^{\text{loc}}(F)$ is regular if and only if* **Brch**$(t)$ is regular.

**Proof.** It is easy to construct a deterministic finite automaton recognizing **Occ**$(f, t)$, **PBrch**$(t)$ or **Brch**$(t)$ from a regular system of equations having $(t, t_2, \ldots, t_n)$ as unique solution, and vice-versa. □

A consequence of this fact is the decidability of the equality of two regular trees defined by regular systems or rational expressions.

An alternative proof has been given in Courcelle et al. [23] and another one can be extracted from Theorem 4.9.2 (see Remark 4.9.10).

## 5. Algebraic trees

This section investigates *algebraic trees*. Such trees are interesting for at least two reasons. They naturally arise in the study of recursive program schemes (modelled after system of mutually recursive functional (i.e. applicative) procedures), when one 'unfolds' the recursion *ad infinitum* in order to characterize by means of a unique infinite tree what in the function defined by a recursive program scheme depends on the interpretation. Another reason is their deep connection with deterministic languages through their branch languages. Whereas 'all properties' of regular trees are decidable, many problems on algebraic trees are undecidable and others are open (in particular the equality problem which is interreducible with the equivalence problem for DPDA's).

As in Section 4, $F$ will denote a fixed ranked alphabet. The extension to a many-sorted alphabet is immediate and need not be done formally.

### 5.1. *Systems of algebraic equations*

In order to define systems of algebraic equations, we shall use the operations of composition and tupling introduced in Section 4.6. Moreover, since we shall only use 1-trees we shall use the notations $T_k$ for $T_{1,k} = M^\infty(F, V_k)$ and $T_{\Omega,k}$ for $M_\Omega^\infty(F, V_k)$. The symbol $\Phi$ will always denote a finite ranked alphabet the elements of which will be used as unknowns in algebraic systems

The set of *scalar monomials of type p over* a ranked alphabet $G$ is the set $SM_p(G)$ of expressions inductively defined as follows ($p \geqslant 0$):

$e \in SM_p(G)$ if and only if
either $e = \pi_{i,p}$ for some $i$ in $[p]$
or $e = f.(e_1, \ldots, e_k)$ for some $f$ in $G_k$ and
some $e_1, e_2, \ldots, e_k$ in $SM_p(G)$.

Hence, for every algebraic theory $A$, if for all $k \geqslant 0$, every $f$ in $G_k$ denotes an element $\nu(f)$ of $A_{1,k}$, then for all $p \geqslant 0$, every *scalar monomial* $e$ in $SM_p(G)$ denotes an element $e_{A,\nu}$ of $A_{1,k}$, inductively defined in an obvious way.

Let $\Phi$ be the ranked alphabet $\{\varphi_1, \ldots, \varphi_n\}$ ($\Phi$ will be so in all this section). *A system of algebraic equations* (or an *algebraic system*) *over F in the set of unknowns* $\Phi$ is a system of the form $\Sigma = \langle \varphi_1 = e_1, \ldots, \varphi_n = e_n \rangle$ where $e_i \in SM_{k_i}(F \cup \Phi)$ and $k_i = \rho(\varphi_i)$ for all $i = 1, \ldots, n$.

A *solution* of $\Sigma$ is an $n$-tuple $(t_1, t_2, \ldots, t_n)$ in $T_{k_1} \times T_{k_2} \times \cdots \times T_{k_n}$ such that $t_i = e_{i T, \nu}$ where $\nu(f) = f(v_1, \ldots, v_{\rho(f)})$ for all $f$ in $F$ and $\nu(\varphi_j) = t_j$ for all $j$ in $[n]$.

Such a system is in *Greibach normal form* if the left-most symbol of each $e_i$ is in $F$.

An alternative way of writing a system $\Sigma$ as above (used for instance in many works on recursive program schemes [15, 18, 22, 24, 25, 27, 37, 44, 53] is

$$\langle \varphi_1(v_1, \ldots, v_{k_1}) = u_1, \ldots, \varphi_n(v_1, \ldots, v_{k_n}) = u_n \rangle$$

where $u_i$ is the element of $M(F \cup \Phi, V_{k_i})$ denoted by $e_i$ in the algebraic theory $M_{F \cup \Phi}$. In that case, a solution of $\Sigma$ is defined as an $n$-tuple $(t_1, \ldots, t_n)$ in $M^\infty(F, V_{k_1}) \times \cdots \times M^\infty(F, V_{k_n})$ such that $t_i = u_i\{t_1/\varphi_1, \ldots, t_n/\varphi_n\}$.

**Example 5.1.1.** Here are two notations for the same algebraic equation:

$$\varphi = c.(\pi_1, \pi_2, \varphi.(\pi_1, h.\pi_2)),$$

$$\varphi(v_1, v_2) = c(v_1, v_2, \varphi(v_1, h(v_2))).$$

The solution $t$ (it is actually unique) is depicted in Fig. 3.

**Theorem 5.1.2.** *An algebraic system* $\Sigma = \langle \varphi_1 = e_1, \ldots, \varphi_n = e_n \rangle$, $e_i \in SM_{k_i}(F \cup \Phi)$ *has a least solution in* $T_{\Omega,k_1} \times \cdots \times T_{\Omega,k_n}$. *If* $\Sigma$ *is in Greibach normal form, it has a unique solution which belongs to* $T_{k_1} \times \cdots \times T_{k_n}$ *and is also its least solution in* $T_{\Omega,k_1} \times \cdots \times T_{\Omega,k_n}$.

**Proof.** Let $E_\Omega$ be the $\omega$-complete partial order $T_{\Omega,k_1} \times \cdots \times T_{\Omega,k_n}$ (its least element is $(\Omega, \Omega, \ldots, \Omega)$) and $|\Sigma|: E_\Omega \to E_\Omega$ be the mapping such that $|\Sigma|(w_1, \ldots, w_n) = (w'_1, \ldots, w'_n)$ with $w'_i = u_i\{w_1/\varphi_1, \ldots, w_n/\varphi_n\}$ for $i = 1, \ldots, n$. This mapping is monotone and $\omega$-continuous by Lemma 3.5.1 hence has a least fix-point $(t_1, \ldots, t_n)$ in $E_\Omega$ which is the least solution of $\Sigma$.

Let now $E$ be the complete metric space $T_{k_1} \times \cdots \times T_{k_n}$. The restriction of $|\Sigma|$ to $E$ is a contracting mapping: $E \to E$ by Lemma 3.5.1 and since $\Sigma$ has been assumed in Greibach normal form. This mapping has a unique fix-point $(t'_1, \ldots, t'_n)$ in $E$ which is the unique solution of $\Sigma$ (this proof technique is used in Bloom [5] and in a more general situation in Arnold and Nivat [2]).

Remark now that if we consider $\Omega$ as an ordinary constant then $E_\Omega$ is also a complete metric space and $|\Sigma|$ has a unique solution in $E_\Omega$. Since $E \subseteq E_\Omega$ the solutions of $|\Sigma|$ in $E$ and $E_\Omega$ are the same and $(t_1, \ldots, t_n) = (t'_1, \ldots, t'_n)$. □

**Remarks 5.1.3.** (1) An algebraic system such that $\rho(\varphi_i) = 0$ for all $i = 1, \ldots, n$ is regular.

A regular system is in Greibach normal form it and only if it is when considered as an algebraic system. The same will hold for properness defined below.

(2) Bloom [5] characterizes all solutions of systems $\Sigma$ which are not in Greibach normal form, (some of them having several solutions).

## 5.2. *The iterative theory of algebraic trees*

An *algebraic tree* is a tree in $M^\infty(F, V_k)$ which is either $v_i$ or a component of the unique solution of an algebraic system in Greibach normal form.

Let $\Sigma$ be a system in Greibach normal form; we shall always denote by $(t_1, \ldots, t_n)$ its unique solution and by $\theta$ (or by $\theta_\Sigma$ if necessary) the second order substitution of $t_1$ for $\varphi_1, \ldots, t_n$ for $\varphi_n$.

It is easy to show that a tree $t$ in $M^\infty(F, V_k)$ is algebraic if and only if $t = \theta_\Sigma(u)$ for some algebraic system $\Sigma$ and some $u$ in $M(F \cup \Phi, V_k)$, where $\Phi$ is the set of unknowns of $\Sigma$.

We denote by $A(F, V_k)$ the set of algebraic trees belonging to $M^\infty(F, V_k)$, by $A(F, V)$ the set $\bigcup_{k \geq 0} A(F, V_k)$ and by $A(F)$ the set of algebraic trees which are in $M^\infty(F)$.

It can be shown that $A(F, V) = A(F \cup V)$ (in the latter notation we consider $V$ as a set of constants).

It can also be shown that $A_\Omega(F, V) = \bigcup_{k \geq 0} A(F \cup \{\Omega\}, V_k)$ is the set of components of least solutions of arbitrary algebraic systems (the proof is similar to the one of Theorem 4.10.1 for regular systems).

Let $A = (A_{n,p})_{n,p \geq 0}$ such that $A_{n,p} = A(F, V_p)^n$. We shall prove that $A$ is an iterative theory. We shall denote it by $A_F$ if it is necessary to specify the ranked alphabet $F$.

**Proposition 5.2.1.** *Let $s_0, s_1, \ldots, s_k$ be algebraic trees, $s_0$ in $A(F, V_k)$, $s_1, \ldots, s_k$ in $A(F, V_p)$. The trees $s_0[s_1, \ldots, s_k]$ and* **Star**$(s_0)$ *(if $s_0 \neq v_1$) are algebraic.*

**Proof.** The proof being trivial if $s_0 \in V_k$ we can exclude this case. Let us also assume that $s_1, \ldots, s_k \notin V_p$.

Without loss of generality, we can assume that $s_0, s_1, \ldots, s_k$ are the first $k+1$ components of the unique solution of a system $\Sigma = \langle \varphi_i(v_1, \ldots, v_{k_i}) = u_i ; 0 \leq i \leq n \rangle$ in Greibach normal form (with $k_0 = k$ and $k_i = p$ for $i = 1, \ldots, k$).

Let us define $\Sigma'$ by adding to $\Sigma$ the new equation

$$\psi(v_1, \ldots, v_p) = u_0[\varphi_1(v_1, \ldots, v_p), \ldots, \varphi_k(v_1, \ldots, v_p)]. \tag{1}$$

Hence $\Sigma'$ is an algebraic system in Greibach normal form having the solution

$$(s_0[s_1, \ldots, s_k], s_0, s_1, \ldots, s_k, \ldots, s_n)$$

where $s_0[s_1, \ldots, s_k]$ corresponds to $\psi$ and $s_i$ to $\varphi_i$ for $i = 0, \ldots n$. Hence $s_0[s_1, \ldots, s_k]$ is algebraic.



If some of the $s_i$'s are in $V_p$, the corresponding $\varphi_i$'s are missing in $\Sigma$ and we define $\Sigma'$ similarly with help of $s_i$ instead of $\varphi_i(v_1, \ldots, v_p)$ in (1).

In order to show that **Star**$(s_0)$ is algebraic, we define $\Sigma''$ by adding to $\Sigma$ the new equation

$$\theta(v_1, \ldots, v_{k-1}) = u_0[\theta(v_1, \ldots, v_{k-1}), v_1, \ldots, v_{k-1}].$$

Hence

$$(\mathbf{Star}(s_0), s_0, s_1, \ldots, s_n)$$

is clearly the solution of $\Sigma''$ (also in Greibach normal form) hence **Star**$(s_0) \in A(F, V_{k-1})$. □

**Corollary 5.2.2.** *$A$ is an iterative theory.*

**Proof.** The first part of Proposition 5.2.1 shows that $A$ is a subtheory of $T$ (as an algebraic theory). The second part shows that $A$ is closed under scalar iteration, hence $A$ is iterative by Proposition 4.7.2. □

The iterative theory $A$ is also investigated by Gallier [38].

Let us mention that in Ginali [40] 'algebraic tree' is just another terminology for 'regular tree'.

The proper inclusion

$$R(F, V) \subset A(F, V)$$

follows from Example 5.1.1: it is clear that the tree $t$ defined there has infinitely many different subtrees; it is algebraic but not regular.

**Proposition 5.2.3.** *The family of algebraic trees is closed under second-order substitution. More precisely, if $\theta$ is the second-order substitution associated with a mapping $\nu : F \to A(G, V)$, then for all $t$ in $A(F, V)$, $\theta(t)$ is an algebraic tree in $A_\Omega(G, V)$ (in $A(G, V)$ if $\theta$ is non erasing).*

**Proof.** The case $t \in V$ is trivial and we exclude it.

The general case easily reduces to the special case $F \cap G = \emptyset$. So we assume this.

Let $\Sigma = \langle \varphi_i(v_1, \ldots, v_{k_i}) = u_i;\ 1 \leq i \leq n \rangle$ be an algebraic system with least solution $(t_1, \ldots, t_n)$ such that $t = t_1$; we let $\Phi = \{\varphi_1, \ldots, \varphi_n\}$.

Let $\Sigma' = \langle \psi_i(v_1, \ldots, v_{h_i}) = w_i;\ 1 \leq i \leq m+p \rangle$ be another system such that its least solution is $(s_1, \ldots, s_{m+p})$ with $s_i = \nu(f_i)$ for $i = 1, \ldots, m$. (We assume that $F = \{f_1, \ldots, f_m\}$ and that $f_i$ and $\psi_i$ are the same symbol.)

Let $\Sigma'' = \Sigma \cup \Sigma'$. Its set of unknowns is $\Phi \cup F \cup \{\psi_{m+1}, \ldots, \psi_{m+p}\}$. Let $(t'_1, \ldots, t'_n, s'_1, \ldots, s'_{m+p})$ be its least solution. We want to show that $t'_1 = \theta(t_1)$. We shall prove in fact that this solution coincides with $(\theta(t_1), \ldots, \theta(t_n), s_1, \ldots, s_{m+p})$.

Let us first consider the special case where $\theta$ is nonerasing and where we can assume that $\Sigma$ and $\Sigma'$ are in Greibach normal form. One can show that $\Sigma''$ has a unique solution although it is not in Greibach normal form.

Hence it suffices to verify that the latter $(n+m+p)$-tuple is a solution of $\Sigma''$, i.e. in fact that its equations from $\Sigma$ are verified (this holds by definition for those from $\Sigma'$).

From $t_i = u_i\{t_1/\varphi, \ldots, t_n/\varphi_n\}$ we get

$$\begin{aligned} \theta(t_i) &= t_i\{\nu(f_1)/f_1, \ldots, \nu(f_m)/f_m\} \\ &= u_i\{t_1/\varphi_1, \ldots, t_n/\varphi_n\}\{\nu(f_1)/f_1, \ldots, \nu(f_m)/f_m\} \\ &= u_i\{\theta(t_1)/\varphi_1, \ldots, \theta(t_n)/\varphi_n, \nu(f_1)/f_1, \ldots, \nu(f_m)/f_m\} \qquad (1) \end{aligned}$$

by the analogous for second-order substitutions of Proposition 3.4.2, and this is exactly what was to be proved since $\psi_{m+j}$, $1 \leq j \leq p$ does not occur in the $u_i$'s.

In order to deal with least solutions, we apply Lemma 2.3.2 to $\Sigma''$ by solving globally the equations giving $(s_1, \ldots, s_{m+p}) = (s'_1, \ldots, s'_{m+p})$ and taking the solution into the remaining ones, namely those from $\Sigma$. Solving these new equations (they form an algebraic system with infinite handsides but we have not introduced them formally) can be done by taking the least upper bound of the sequence $S^i(\Omega, \ldots, \Omega)$ where $S$ is a mapping: $M_\Omega^\infty(G, V_k) \times \cdots \times M_\Omega^\infty(G, V_{k_n})$ into itself which is derived from $|\Sigma''|$ as follows:

$S(z_1, \ldots, z_n)$ consists in the first $n$ components of $|\Sigma''|(z_1, \ldots, z_n, s_1, \ldots, s_{m+p})$ for $z_i$ in $M_\Omega^\infty(G, V_{k_i})$, $1 \leq i \leq n$.

Note that for $b_1, \ldots, b_n$ in $M_\Omega^\infty(F, V_{k_1}), \ldots, M_\Omega^\infty(F, V_{k_n})$:

$$S(\theta(b_1), \ldots, \theta(b_n)) = |\Sigma''|(\theta(b_1), \ldots, \theta(b_n), s_1, \ldots, s_{m+p})$$
restricted to its first $n$ components

$$= (|\Sigma|(b_1, \ldots, b_n))\{s_i/f_i;\ 1 \leq i \leq m+p\}$$

$$= \theta(|\Sigma|(b_1, \ldots, b_n))$$
since the calculation step (1) above holds for arbitrary $b_1, \ldots, b_n$ in place of $t_1, \ldots, t_n$.

By using an obvious vector notation,

$$S(\theta(\boldsymbol{b})) = \theta(|\Sigma|(\boldsymbol{b}))$$

whence

$$\theta(|\Sigma|^i(\boldsymbol{\Omega})) = S^i(\theta(\boldsymbol{\Omega})) = S^i(\boldsymbol{\Omega})$$

since

$$\theta(\boldsymbol{\Omega}) = \boldsymbol{\Omega}.$$

Since $\theta$ is $\omega$-continuous,

$$\theta(t_1, \ldots, t_n) = \theta(\mathbf{Sup}|\Sigma|^i(\boldsymbol{\Omega}))$$

$$= \mathbf{Sup}_{i \geq 0}(S^i(\boldsymbol{\Omega}))$$

$$= (t'_1, \ldots, t'_n).$$

Hence we have obtained $t'_1 = \theta(t_1)$ as desired. □

### 5.3. *Algebraic trees and schematic tree languages*

We state that the components of the least solution $(t_1, \ldots, t_n)$ of an algebraic system $\Sigma$ can be defined as the least upper bounds of directed sets of trees $L_1, \ldots, L_n$ generated from a context-free tree grammar $\Sigma_\Omega$ associated with $\Sigma$.

Let $\Sigma = \langle \phi_1(v_1, \ldots, v_{k_1}) = u_1, \ldots, \phi_n(v_1, \ldots, v_{k_n}) = u_n \rangle$. Let $\Sigma_\Omega$ be the set of pairs $(\phi_i(v_1, \ldots, v_{k_i}), u_i)$ and $(\phi_i(v_1, \ldots, v_{k_i}), \Omega)$ and let $\xrightarrow{*}_{\Sigma_\Omega}$ be the semi-Thue relation on $M_\Omega(F \cup \Phi, V)$ associated with $\Sigma_\Omega$.

The triple $(F, \Phi, \Sigma_\Omega)$ is in fact a *context-free tree-grammar* (see Engelfriet and Schmidt [35] for a detailed study) of a special form: we call it a *schematic* tree-grammar as in Courcelle [15] (since it comes from a recursive program scheme).

For every $u$ in $M(F \cup \Phi, V_k)$, the schematic grammar $\Sigma_\Omega$ generates a tree-language $L(u, \Sigma_\Omega) = \{w \in M_\Omega(F, V_k) \mid u \xrightarrow{*}_{\Sigma_\Omega} w\}$. Such a tree-language is called a *schematic* tree-language.

**Lemma 5.3.1** (Nivat [53]). *$L(u, \Sigma_\Omega)$ is directed with respect to $\leq$.*

Hence $L(u, \Sigma_\Omega)$ has a least upper bound that we shall denote by $\tau(u)$.

Let us denote by $\theta$ the second order substitution of $t_1$ for $\phi_1, \ldots, t_n$ for $\phi_n$ where $(t_1, \ldots, t_n)$ is here the *least* solution of $\Sigma$ in $T_{\Omega,k_1} \times \cdots \times T_{\Omega,k_n}$.

The following result is often referred to as Schützenberger's theorem (by reference to a similar result of [64]):

**Theorem 5.3.2.** (1) *The n-tuple of trees $(\tau(\phi_1(v_1, \ldots, v_{k_1})), \ldots, \tau(\phi_n(v_1, \ldots, v_{k_n})))$ is the least solution of $\Sigma$.*

(2) *For all $u$ in $M(F \cup \Phi, V)$, $\tau(u) = \theta(u)$.*

**Proof.** (1) We only sketch the proof. Let $L_i = L(\phi_i(v_1, \ldots, v_{k_i}), \Sigma_\Omega)$ and $\tau_i = \mathbf{Sup}(L_i)$ for $i = 1, \ldots, n$. Let us show that

$$u_i\{\tau_1/\phi_1, \ldots, \tau_n/\phi_n\} \leq \tau_i \quad \text{for all } i. \tag{*}$$

Since the second-order substitution is $\omega$-continuous in all its arguments, it suffices to show that $u_i\{w_1/\phi_1, \ldots, w_n/\phi_n\} \leq \tau_i$ for all $w_1 \in L_1, \ldots, w_n \in L_n$ and $i \in [n]$. It can be shown that for all $u \in M(F \cup \Phi, V)$, all $w_1 \in L_1, \ldots, w_n \in L_n$, $u\{w_1/\phi_1, \ldots, w_n/\phi_n\} \in L(u, \Sigma_\Omega)$ (by induction on the structure of $u$). Hence (*) is established and one can conclude from Theorem 2.3.1 that $t_i \leq \tau_i$ for all $i$.

The other direction is more technical. One can establish that $\tau_i \leq t_i$ by showing that for all $w_i$ in $L_i$ there exists $j$ such that $w_i \leq t_i^{(j)}$ where $(t_1^{(j)}, \ldots, t_n^{(j)}) = |\Sigma|^{(j)}(\Omega, \Omega, \ldots, \Omega)$ (see the proof of Theorem 5.1.2). One can take $j$ equal to the length of a derivation $\phi_i(v_1, \ldots, v_{k_i}) \xrightarrow{*}_{\Sigma_\Omega} w_i$ (see Nivat [53]).

Hence one can conclude that $(t_1, \ldots, t_n) = (\tau_1, \ldots, \tau_n)$.

(2) For $u$ in $M(F \cap \Phi, V_k)$ the equality $\theta(u) = \tau(u)$ follows from the above result applied to a system made of one new equation $\phi(v_1, \ldots, v_k) = u$ and all the equations of $\Sigma$. □

### 5.4. *Normal forms and reductions*

It is not very difficult to prove that a system can be put in *Greibach normal form* provided its least solution has no occurrence of the symbol $\Omega$.

We shall also prove that the number of variables occurring in the left-hand sides of equations can be reduced in a way which eliminates useless variables.

Let $\Sigma$ and $(t_1, \ldots, t_n)$ be as in Section 5.3. The system $\Sigma$ is *proper* if $t_i \neq \Omega$ for all $i = 1, \ldots, n$. Since $u_1, \ldots, u_n$ have no occurrence of $\Omega$, we have the following:

**Proposition 5.4.1.** *The following properties of an algebraic system $\Sigma$ as above are equivalent*:

(i) *$\Sigma$ is proper,*
(ii) *$t_i \in M^\infty(F, V)$ for all $i = 1, \ldots, n$,*
(iii) *$\Sigma$ has a unique solution in $T_{\Omega,k_1} \times \cdots \times T_{\Omega,k_n}$,*
(iv) *$\Sigma$ has a unique solution in $T_{k_1} \times \cdots \times T_{k_n}$.*

*These conditions are decidable.*

**Proof.** (ii) ⇒ (iii): any other solution $(t'_1, \ldots, t'_n)$ satisfies $t_i \leq t'_i$ for $i = 1, \ldots, n$. Since $t_1, \ldots, t_n$ have no occurrence of $\Omega$, they are maximal hence $t_i = t'_i$ for all $i$.

(iii) ⇒ (iv) is obvious.

(iv) ⇒ (i). $\Sigma$ is not proper if and only if the set $J = \{i \in [n] \mid t_i = \Omega\}$ is not empty. It can be shown that $J$ is the set of indices $i$ such that

$$\phi_i(v_1, \ldots, v_{k_i}) \xrightarrow[\Sigma]{*} \phi_j(w_1, \ldots, w_{k_j}),$$

$$\phi_j(v_1, \ldots, v_{k_j}) \xrightarrow[\Sigma]{*} \phi_j(w'_1, \ldots, w'_{k_j})$$

for some $j$ in $[n]$, $w_1, \ldots, w_{k_j}, w'_1, \ldots, w'_{k_i}$ in $M(F \cup \Phi, V_{k_i})$. This characterization allows us to construct $J$ and to decide whether $\Sigma$ is proper.

Let $s$ be any element of $M^{\infty}(F)$. It can be shown that there exists a unique solution $(s_1, \ldots, s_n)$ of $\Sigma$ in $T_{k_1} \times \cdots \times T_{k_n}$ such that $s_i = s$ for all $i$ in $J$.

Any other choice of $s$ gives another solution of $\Sigma$ in $T_{k_1} \times \cdots \times T_{k_n}$. Hence if $\Sigma$ satisfies (iv), it must be proper.

(i) $\Rightarrow$ (ii). For any tree $t$ in $M^{\infty}_{\Omega}(F, V)$ let us define $\|t\|$ as $\mathbf{Min}\{|\alpha| \mid \alpha \in \mathbf{Occ}(\Omega, t)\}$ so that $t \in M^{\infty}(F, V)$ if and only if $\|t\| = \infty$.

Let $\Sigma$ be proper; let $w$ be a tree in $M(F \cup \Phi, V)$ such that $\|\theta(w)\|$ is minimal.
If there exist several trees satisfying this, let us select one of minimal size.

*Case 1*: $\|\theta(w)\| = 0$. This is possibly only if $w = \phi_i(w_1, \ldots, w_{k_i})$. Since $t_i \neq \Omega$, this implies $t_i = v_j$ and $\theta(w_j) = \Omega$, i.e. $\|\theta(w_j)\| = 0$ but this contradicts the minimality of $|w|$.

*Case 2*: $\|\theta(w)\| = n > 0$. If $w = f(w_1, \ldots, w_k)$ then $\|\theta(w_j)\| = n - 1$ for some $j$ and this contradicts the minimality of $\|\theta(w)\|$. Hence $w = \phi_i(w_1, \ldots, w_{k_i})$ and as above $t_i \notin V$. Hence $t_i = f(t'_1, \ldots, t'_k)$. This shows that $\phi_i(v_1, \ldots, v_{k_i}) \xrightarrow{+}_{\Sigma} f(u'_1, \ldots, u'_k)$ for some $u'_1, \ldots, u'_k$ in $M(F \cup \Phi, V_{k_i})$. And we have

$$\theta(w) = f(\theta(w'_1), \ldots, \theta(w'_k)),$$

$$w'_i = u'_i[w_1, \ldots, w_{k_i}].$$

But $\|\theta(w'_i)\| = n - 1$ for some $i$ and this contradicts the choice of $w$.

Hence $\|\theta(w)\| = \infty$, i.e. $\theta(w) \in M^{\infty}(F, V_k)$ for all $w$ in $M(F \cup \Phi, V_k)$. This holds in particular for $w = \phi_i(v_1, \ldots, v_{k_i})$ (for which $\theta(w) = t_i$). And this proves (ii). □

**Definition 5.4.2.** Let us recall that $\Sigma$ is in Greibach normal form if its right-hand sides (the $u_i$'s) belong to $F(M(\Phi \cup F, V))$. Note that by introducing extra unknowns (corresponding to the function symbols of $F$), one can put $\Sigma$ in such a form that $u_i \in F(M(\Phi, V))$. This more stringent form corresponds closer to the usual Greibach normal form for context-free grammars.

The algebraic system $\Sigma$ is *V-reduced* if for all $i$ in $[n]$, all $j$ in $[k_i]$, the variable $v_j$ has an occurrence in $t_i$. This means that all variables occurring in $\Sigma$ are 'useful'.

An algebraic system is *trim* if it is $V$-reduced and in Greibach normal form. Our purpose is to show that any proper system can be 'trimmed'.

We need a precise definition. Let $\Sigma' = \langle \psi_1(v_1, \ldots, v_{h_1}) = u'_1, \ldots, \psi_m(v_1, \ldots, v_{h_m}) = u'_m \rangle$ be another algebraic system and $\Psi = \{\psi_1, \ldots, \psi_m\}$. We say that $\Sigma$ is *$\Sigma'$-definable* if there exist $w_1$ in $M(F \cup \Psi, V_{k_1}), \ldots, w_n$ in $M(F \cup \Psi, V_{k_n})$ such that

$$t_i = \theta_{\Sigma'}(w_i) \quad \text{for } i = 1, \ldots, n.$$

In that case $A_{\Sigma}(F, V) \subseteq A_{\Sigma'}(F, V)$, where $A_{\Sigma}(F, V)$ denotes the set of *$\Sigma$-definable trees*, i.e. the set $\{\theta_{\Sigma}(u) \mid u \in M(F \cup \Phi, V)\}$.

Conversely, if $A_{\Sigma}(F, V) \subseteq A_{\Sigma'}(F, V)$ then $\Sigma$ is $\Sigma'$-definable.

We shall also say that $\Sigma$ *is equivalent to* $(\Sigma', w_1, \ldots, w_n)$ or that $(\Sigma', w_1, \ldots, w_n)$ is *a translation* of $\Sigma$.

If $(\Sigma', w_1, \ldots, w_n)$ is a translation of $\Sigma$ and $(\Sigma, w'_1, \ldots, w'_n)$ is a translation of $\Sigma'$ then we say that $\Sigma$ and $\Sigma'$ are *intertranslatable* (and this implies $A_\Sigma(F, V) = A_{\Sigma'}(F, V)$). In the special case where $\Phi = \Psi$, $w_i = w'_i = \phi_i(v_1, \ldots, v_{k_i})$ we shall say that $\Sigma$ and $\Sigma'$ are *equivalent*.

**Proposition 5.4.3.** *Any proper algebraic system is intertranslatable with a trim algebraic system.*

**Proof.** Let $\Sigma = \langle \phi_i(v_1, \ldots, v_{k_i}) = u_i;\ 1 \leq i \leq n\rangle$ be proper and $(t_1, \ldots, t_n)$ be its unique solution.

One can determine the set $I$ of indices $i$ such that $t_i \notin V$. Without loss of generality (and just to simplify the notations), we can assume that $I = \{1, 2, \ldots, l\}$ with $0 \leq l \leq n$ ($I = \emptyset$ if $l = 0$). Let $t_i = v_{h(i)}$ for $i = l+1, \ldots, n$, for some mapping $h : \{l+1, \ldots, n\} \to [n]$.

One can also determine (see Courcelle [15]), for each $i \in I$ the set $H(i)$ of variables occurring in $t_i$. Let us write it $H(i) = \{v_{h(i,1)}, \ldots, v_{h(i,m_i)}\}$.

Let us define $\Psi = \{\psi_i \mid 1 \leq i \leq l\}$ and $\rho(\psi_i) = m_i$ for all $i$. We shall translate $\Sigma$ into $(\Sigma', w_1, \ldots, w_n)$ for some system $\Sigma'$ having $\Psi$ as set of unknowns.

Let us define immediately:

$$w_i = \psi_i(v_{h(i,1)}, \ldots, v_{h(i,m_i)}) \quad \text{for } 1 \leq i \leq l, \tag{1}$$

$$w_i = v_{h(i)} \quad \text{for } l+1 \leq i \leq n. \tag{2}$$

We define $\Sigma' = \langle \Psi_i(v_1, \ldots, v_{m_i}) = u'_i;\ 1 \leq i \leq l\rangle$ by letting $u'_i$ be the unique element of $M(F \cup \Psi, V_{m_i})$ such that

$$u'_i[v_{h(i,1)}, \ldots, v_{h(i,m_i)}] = u''_i\{w_1/\phi_1, \ldots, w_n/\phi_n\} \tag{3}$$

where $u''_i$ is some element of $M(F \cup \Phi, V_{k_i})$ such that $\mathbf{First}(u''_i) \in F$ and $\phi_i(v_1, \ldots, v_{k_i}) \xrightarrow{*}_\Sigma u''_i$.

We have to show that the right-hand side of (3) belongs to $M(F \cup \Phi, H(i))$ in order to ensure the existence of $u'_i$.

It is easy to see that every variable $v_j$ occurring in $u''_i\{w_1/\phi_1, \ldots, w_n/\phi_n\}$ also occurs in $u''_i\{t_1/\phi_1, \ldots, t_n/\phi_n\}$.

Remark now that $t_i = u''_i\{t_1/\phi_1, \ldots, t_n/\phi_n\}$ since $\phi_i(v_1, \ldots, v_{k_i}) \xleftrightarrow{*}_\Sigma u''_i$ and for all $u, u', u \xleftrightarrow{*}_\Sigma u'$ implies $\theta_\Sigma(u) = \theta_\Sigma(u')$.

Hence $v_j$ belongs to $\mathbf{Var}(t_i) = H(i)$.

We shall now prove that $(\Sigma', w_1, \ldots, w_n)$ is a translation of $\Sigma$.

We do this by means of another system in Greibach normal form:

$$\Sigma''' = \langle \phi_i(v_1, \ldots, v_{k_i}) = u'''_i;\ 1 \leq i \leq l\rangle$$

where

$$u'''_i = u''_i\{v_{h(l+1)}/\phi_{l+1}, \ldots, v_{h(n)}/\phi_n\} = u''_i\{t_{l+1}/\phi_{l+1}, \ldots, t_n/\phi_n\}.$$

One has for all $i = 1, \ldots, l$,

$$u_i'''\{t_1/\phi_1, \ldots, t_l/\phi_l\} = u_i''\{t_1/\phi_1, \ldots, t_n/\phi_n\}.$$

Hence $(t_1, \ldots, t_l)$ is *the* solution of $\Sigma'''$. Let us show that $(\theta_{\Sigma'}(w_1), \ldots, \theta_{\Sigma'}(w_l))$ is also a solution of $\Sigma'''$; let $(t_1', \ldots, t_l')$ be the solution of $\Sigma'$; for all $i \in [l]$,

$$\begin{aligned}
\theta_{\Sigma'}(w_i) &= t_i'[v_{h(i,1)}, \ldots, v_{h(i,m_i)}] \\
&= u_i'\{t_1'/\psi_1, \ldots, t_l'/\psi_l\}[v_{h(i,1)}, \ldots, v_{h(i,m_i)}] \\
&= u_i'[v_{h(i,1)}, \ldots, v_{h(i,m_i)}]\{t_1'/\psi_1, \ldots, t_l'/\psi_l\} \\
&= u_i''\{w_1/\phi_1, \ldots, w_n/\phi_n\}\{t_1'/\psi_1, \ldots, t_l'/\psi_l\} \\
&= u_i''\{w_j\{t_1'/\psi_1, \ldots, t_l'/\psi_l\}/\phi_j;\ 1 \leq j \leq n\} \\
&= u_i''\{v_{h(l+1)}/\phi_{l+1}, \ldots, v_{h(n)}/\phi_n\}\{\theta_{\Sigma'}(w_1)/\phi_1, \ldots, \theta_{\Sigma'}(w_l)/\phi_l\} \\
&= u_i'''\{\theta_{\Sigma'}(w_1)/\phi_1, \ldots, \theta_{\Sigma'}(w_l)/\phi_l\}.
\end{aligned}$$

Hence $(\theta_{\Sigma'}(w_i))_{1 \leq i \leq l}$ is a solution of $\Sigma'''$ and $t_i = \theta_{\Sigma'}(w_i)$ for all $i$ in $[l]$.

Since $\theta_{\Sigma'}(w_i) = v_{h(i)}$ for all $i \in \{l+1, \ldots, n\}$, we have shown that $(\Sigma', w_1, \ldots, w_n)$ is a translation of $\Sigma$.

On the other hand

$$t_i' = \theta_{\Sigma}(w_i') \quad \text{for } i = 1, \ldots, l$$

where

$$w_i' = \phi_i(v_1, \ldots, v_{k_i})[v_1/v_{h(i,1)}, \ldots, v_{m_i}/v_{h(i,m_i)}]$$

(this follows from (1), (2) and the fact that $\mathbf{Var}(t_i) = H(i)$) hence $\Sigma$ and $\Sigma'$ are intertranslatable. $\square$

We shall use this construction to decide whether an algebraic tree is locally finite, by means of another proposition.

**Proposition 5.4.4.** *Let $F$ have no constant. Let $\Sigma$ be a trim algebraic system over $F$, $\Phi$ its set of unknowns and $(t_1, \ldots, t_n)$ its unique solution. The following properties are equivalent*:

(1) $\rho(\phi) \neq 0$ *for all* $\phi$ *in* $\Phi$,

(2) $t_i \in M^{\mathrm{loc}}(F, V)$ *for all* $i = 1, \ldots, n$,

(3) $A_\Sigma(F, V) \subseteq M^{\mathrm{loc}}(F, V)$.

**Proof.** (3) $\Rightarrow$ (2) is trivial since $t_i \in A_\Sigma(F, V)$.

(2) $\Rightarrow$ (1) is clear since $t_i \in M^\infty(F, V)$: if $t_i$ is locally finite $\mathbf{Var}(t_i) \neq \emptyset$ hence $\rho(\phi_i) \neq 0$.

(1) $\Rightarrow$ (3). Let $u \in M(F \cup \Phi, V)$ and $t = \theta_\Sigma(u)$. By Proposition 5.6.1 to be proved later, for all $\alpha$ in $\mathbf{Dom}(t)$, there exists $u'$ in $M(F \cup \Phi, V)$ such that $t/\alpha = \theta_\Sigma(u')$. Note that $\mathbf{Var}(u') \neq \emptyset$.

Since $\Sigma$ is trim, $\mathbf{Var}(\theta_\Sigma(u')) = \mathbf{Var}(u')$. Hence $\mathbf{Var}(t/\alpha) \neq \emptyset$ and this shows that $t$ is locally finite. □

**Corollary 5.4.5.** *It is possible to decide whether a given algebraic tree is locally finite.*

**Proof.** Let $\Sigma$ be an algebraic system over $F$ and $t = \theta_\Sigma(u)$ for $u$ in $M(F \cup \Phi, V)$.

One can translate $\Sigma$ into $(\Sigma', w_1, \ldots, w_n)$ where $\Sigma'$ is an algebraic system over $F$ without constants. It suffices to define a new variable $v_a$ for each $a$ in $F_0$ and add $v_a$ to the variable list of $\phi_i$ if $a$ occurs in $t_i$. We omit the details.

Hence $t = \theta_{\Sigma'}(u')$ for some $u' \in M(F \cup \Phi', V)$.

Let us translate $\Sigma'$ into a trim system $\Sigma''$ by using Proposition 5.4.4 and let us restrict $\Sigma''$ to the equations that are really useful for the definition of $t$.

This means that $t = \theta_{\Sigma''}(u'')$ for some trim algebraic system $\Sigma''$ with set of unknowns $\Phi''$ and such that for all $\phi$ in $\Phi''$, there exists $\alpha$ in $\mathbf{Dom}(t)$ and $w''$ in $M(F \cup \Phi'', V)$ such that

$$t/\alpha = \theta_{\Sigma''}(w''), \qquad \mathbf{First}(w'') = \phi.$$

From this one can deduce that $t$ is locally finite if and only if $\Sigma''$ satisfies condition (2) of Proposition 5.4.4, i.e. if and only if $\rho(\phi) \neq 0$ for all $\phi$ in $\Phi''$. This is decidable. □

### 5.5. *Algebraic trees and deterministic languages*

The following theorem draws a bridge between algebraic trees and deterministic context-free languages. Let us recall that these languages can be defined by deterministic pushdown automata (DPDA's) or equivalently, by LR($k$) or strict deterministic grammars (see Harrison [45]).

The equivalence problem for DPDA's, i.e. the problem of deciding whether two DPDA's $A_1$ and $A_2$ define the same language is open. Many decidable subcases have been discovered (Valiant [68], Oyamaguchi et al. [58, 59] in particular).

By using the notations of Section 1.6

**Theorem 5.5.1.** (1) *A tree in $M^{\mathrm{loc}}(F)$ is algebraic if and only if* $\mathbf{Brch}(t)$ *is a deterministic language.*

(2) *A tree $t$ in $M^\infty(F)$ is algebraic if and only if* $\mathbf{PBrch}(t)$ *(or $L(t)$) is a deterministic language.*

(3) *If a tree $t$ in $M^\infty(F)$ is algebraic then* $\mathbf{Occ}(f, t)$ *is a deterministic language for all $f$ in $F$.*

Part (1) is proved in Courcelle [15], part (2) follows easily and part (3) is proved in Gallier [37]. The proofs are much too technical to be even sketched here.

This result is fully similar to Theorem 4.11.1 concerning regular trees, except that the converse to (3) yields an open problem.

**Open problem 5.5.2.** *Is it true that a tree t in $M^\infty(F)$ such that F is finite and* **Occ**$(f, t)$ *is a deterministic language for all f in F is algebraic?*

The answer is yes if $F$ consists of two symbols of the same arity. This is due to the fact that the complement of a deterministic language $L$ can be recognized by the same automaton as $L$ except for accepting modes.

We do not make any conjecture concerning the general case but we give an equivalent formulation:

*Is it true that if $(L_1, L_2, \ldots, L_n)$ is a partition of $X^*$ into n deterministic languages then the language $L_1 1 \cup L_2 2 \cup \cdots \cup L_n n$ over $X \cup [n]$ is deterministic?*

**Theorem 5.5.3.** *The equivalence problem for DPDA's and the equality problem for algebraic trees are interreducible.*

The reduction from algebraic trees to DPDA's follows from the remark that $t = t'$ if and only if **Occ**$(f, t)$ = **Occ**$(f, t')$ for all $f$ in $F$ and part (2) of Theorem 5.5.1. It can also be established by means of **PBrch**$(t)$ or **Brch**$(t)$ (Courcelle [15]).

The other reductions are much more technic , they are proved in Courcelle [15] and Gallier [37].

**Consequences 5.5.4.** The above cited constructions yield the following facts:

(1) Every decidable case of the equivalence problem for DPDA's yields decidable cases of the equality problem for algebraic trees. (Not just *one* case because there exist several reductions of the equality problem for algebraic trees to the equivalence problem for DPDA's: two by Courcelle and one by Gallier [14, 15, 37]. Actually it is not at all easy to have handy characterizations of the corresponding classes of algebraic systems. But this is a direction for future research.

(2) Every decidable case of the equality problem for algebraic trees yields decidable cases of the equivalence problem for DPDA's. This is also a largely open research direction.

**Remark 5.5.5.** Since **Brch**$(t)$ can be defined by a grammar, **Var**$(t)$ can be computed. One can decide whether a given system is trim.

*5.6. Finding one's way in an algebraic tree*

We shall describe an automaton the states of which are trees which describes the paths from the root to any node in an algebraic tree.

It will allow us to determine the subtree issued from any node of an algebraic tree.

Let $\Sigma = \langle \phi_i(v_1, \ldots, v_{k_i}) = u_i;\ 1 \leq i \leq n \rangle$ be an algebraic system in Greibach normal form. We associate with $\Sigma$ a partial mapping:

$$\gamma_\Sigma : \mathbf{N}_+ \times M(F \cup \Phi, V_k) \to M(F \cup \Phi, V_k)$$

and its canonical extension $\gamma_\Sigma^*$ to $\mathbf{N}_+^*$, by the following definitions ($\Sigma$ being fixed we use $\gamma$ and $\gamma^*$ instead of $\gamma_\Sigma$ and $\gamma_\Sigma^*$):
- $\gamma(i, t) = s$ if and only if
  (1) either $t = f(s_1, \ldots, s_k)$, $1 \leq i \leq k$ and $s = s_i$,
  (2) or $t = \phi_i(s_1, \ldots, s_{k_i})$, $u_i$ is of the form $f(u'_1, \ldots, u'_k)$ and $s = u'_i[s_1, \ldots, s_{k_i}]$ with $1 \leq i \leq k$,
- $\gamma^*(\varepsilon, t) = t$ for all $t$ in $M(F \cup \Phi, V_k)$,
- $\gamma^*(i\alpha, t) = \gamma^*(\alpha, \gamma(i, t))$ for $i \in \mathbf{N}_+$, $\alpha \in \mathbf{N}_+^*$ provided $\gamma(i, t)$ and $\gamma^*(\alpha, \gamma(i, t))$ are defined; otherwise, $\gamma^*(i\alpha, t)$ is undefined.

**Proposition 5.6.1.** *Let $u \in M(F \cup \Phi, V_k)$ and $\alpha \in \mathbf{N}_+^*$. Then $\alpha \in \mathbf{Dom}(\theta_\Sigma(u))$ if and only if $\gamma_\Sigma^*(\alpha, u)$ is defined. If it is then $\theta_\Sigma(u)/\alpha = \theta_\Sigma(\gamma_\Sigma^*(\alpha, u))$.*

**Proof.** An easy induction on $|\alpha|$. □

Let $\bar{\gamma}_\Sigma^* : \mathbf{N}_+^* \times M(F \cup \Phi, V_k) \to F \cup V_k$ be defined as follows (we use $\bar{\gamma}^*$ instead of $\bar{\gamma}_\Sigma^*$):
- $\bar{\gamma}^*(\alpha, t) = \mathbf{First}(\gamma^*(\alpha, t))$ if $\gamma^*(\alpha, t)$ is defined and belongs to $F(M(F \cup \Phi, V_k)) \cup V_k$,
- $\bar{\gamma}^*(\alpha, t) = f$ if $\gamma^*(\alpha, t) = \phi_i(w_1, \ldots, w_{k_i})$ and $f = \mathbf{First}(u_i)$
- $\bar{\gamma}^*(\alpha, t)$ is undefined if $\gamma^*(\alpha, t)$ is.

Hence, for all $u$ in $M(F \cup \Phi, V)$, all $\alpha$ in $\mathbf{Dom}(\theta_\Sigma(u))$ we have

$$\theta_\Sigma(u)(\alpha) = \bar{\gamma}^*(\alpha, u). \quad \square$$

**Corollary 5.6.2.** *For $u, u'$ in $M(F \cup \Phi, V)$, $\theta_\Sigma(u) = \theta_\Sigma(u')$ if and only if the mappings $\lambda\alpha \in \mathbf{N}_+^* . \bar{\gamma}^*(\alpha, u)$ and $\lambda\alpha \in \mathbf{N}_+^* . \bar{\gamma}^*(\alpha, u')$ are equal.*

### *5.7. Two congruences associated with an algebraic system*

Let $\Sigma$, $(t_1, \ldots, t_n)$, $\theta$ be as in Section 5.6. Let us also assume that $\Sigma$ is $V$-reduced. Hence $\Sigma$ is trim.

Since $\Sigma$ is in Greibach normal form, the second-order substitution $\theta$ is continuous and extends uniquely to $M^\infty(F \cup \Phi, V)$ by Proposition 3.5.6.

There corresponds to $\Sigma$ a congruence $\overset{*}{\leftrightarrow}_\Sigma$ on $M(F \cup \Phi, V)$ generated by $\Sigma$ considered as a set of pairs of terms, and a congruence on $M^\infty(F \cup \Phi, V)$ defined by

$$t \equiv_\Sigma t' \text{ if and only if } \theta(t) = \theta(t').$$

In the following theorem, we compare $\overset{*}{\leftrightarrow}_\Sigma$ with the restriction of $\equiv_\Sigma$ to $M(F \cup \Phi, V)$, also denoted by $\equiv_\Sigma$.

A congruence $\sim$ on $M(G, V)$ is *stable* if $\sigma(t) \sim \sigma(t')$ for all finite first-order substitution and all $t, t'$ such that $t \sim t'$.

**Theorem 5.7.1.** (1) *$\overset{*}{\leftrightarrow}_\Sigma$ and $\equiv_\Sigma$ are stable congruences on $M(F \cup \Phi, V)$ and $\overset{*}{\leftrightarrow}_\Sigma \subset \equiv_\Sigma$.*

(2) $\equiv_\Sigma$ *is simplifiable.*
(3) $\overset{*}{\leftrightarrow}_\Sigma$ *is semi-decidable and* $\equiv_\Sigma$ *is semi-refutable.*
(4) $t \equiv_\Sigma t'$ *if and only if*

$$\mathbf{Inf}\{d(u, u') \mid t \underset{\Sigma}{\overset{*}{\leftrightarrow}} u, t' \underset{\Sigma}{\overset{*}{\leftrightarrow}} u', u, u' \in M(F \cup \Phi, V)\} = 0.$$

Before starting the proof, let us make some remarks:

**Remark 5.7.2.** (1) Proving that $\equiv_\Sigma$ is semi-decidable is equivalent to proving that the equivalence problem for DPDA's is solvable.

(2) Here is an example of $\Sigma$ such that $\overset{*}{\leftrightarrow}_\Sigma$ and $\equiv_\Sigma$ are not the same:

$$\Sigma = \langle \phi(v_1) = f(v_1, \phi\phi v_1), \psi(v_1) = f(\phi v_1, \psi\phi v_1)\rangle,$$

$$\phi\phi v_1 \equiv_\Sigma \psi v_1,$$

$$\phi\phi v_1 \underset{\Sigma}{\overset{*}{\not\leftrightarrow}} \psi v_1 \quad (\text{if } \psi v_1 \underset{\Sigma}{\overset{*}{\leftrightarrow}} u, \text{ then } u \text{ has occurrences of } \psi).$$

(3) It is shown in Courcelle and Vuillemin [27] that every system $\Sigma$ such that $\rho(\phi_i) = 1$ for all $i$ and which is $V$-reduced and in Greibach normal form can be transformed into an equivalent system $\Sigma'$ (on the same set $\Phi$ of unknowns and having the same solution) such that $\overset{*}{\leftrightarrow}_{\Sigma'}$, $\equiv_{\Sigma'}$ (and $\equiv_\Sigma$) are the same.

(4) Whether $\overset{*}{\leftrightarrow}_\Sigma$ is decidable in general is an open question raised by R. Milner. A similar congruence on free monoids has been shown undecidable in general by Book [12]. But this does not prove that $\overset{*}{\leftrightarrow}_\Sigma$ is.

**Proof of Theorem 5.7.1.** (1) That $\overset{*}{\leftrightarrow}_\Sigma$ and $\equiv_\Sigma$ are congruences is obvious. Since $\Sigma$ considered a binary relation on $M(F \cup \Phi, V)$ is included in $\equiv_\Sigma$, so is $\overset{*}{\leftrightarrow}_\Sigma$ the least congruence containing $\Sigma$.

(2) Let $w = a(w_1, \ldots, w_k) \equiv_\Sigma a(w'_1, \ldots, w'_k) = w'$.
If $a = f \in F$ then $f(\theta(w_1), \ldots, \theta(w_k)) = f(\theta(w'_1), \ldots, \theta(w'_k))$, hence $\theta(w_i) = \theta(w'_i)$, i.e. $w_i \equiv_\Sigma w'_i$ for $i = 1, \ldots, k$. If $a = \phi_i \in \Phi$ then $t_i[\theta(w_1), \ldots, \theta(w_k)] = t_i[\theta(w'_1), \ldots, \theta(w'_k)]$. Since $\Sigma$ has been assumed $V$-reduced each $v_i$, $i = 1, \ldots, k$ occurs in $t_i$, hence $\theta(w_i) = \theta(w'_i)$ for all $i = 1, \ldots, k$ by Proposition 3.4.1.

(3) As any finitely generated congruence for which one knows a finite set of generators, $\overset{*}{\leftrightarrow}_\Sigma$ is semi-decidable.

Since $\bar{\gamma}^*$ is computable, Corollary 5.6.2 shows that $\equiv_\Sigma$ is semi-refutable, i.e. that its negation is semi-decidable.

(4) Let $t \not\equiv_\Sigma t'$. Let us consider the mapping $|\Sigma| : E \to E$ where $E = M(F \cup \Phi, V_{k_1}) \times \cdots \times M(F \cup \Phi, V_{k_n})$ which was used in Theorem 5.1.2; we recall its definition:

$$|\Sigma|(w_1, \ldots, w_n) = (w'_1, \ldots, w'_n),$$

$$w'_i = u_i\{w_1/\phi_1, \ldots, w_n/\phi_n\} \quad \text{for all } i \text{ in } [n].$$

Since $\Sigma$ is assumed in Greibach normal form, this mapping is contracting and $|\Sigma|^m(\phi_1(v_1,\ldots,v_{k_1}),\ldots,\phi_n(v_1,\ldots,v_k))=(w_1^m,\ldots,w_n^m)$ converges (but not in $E$) to $(t_1,\ldots,t_n)$, the unique solution of $\Sigma$.

In particular,

$$\delta(t_i, w_i^m) \geq m \quad \text{for all } i=1,\ldots,n \text{ and } m\in\mathbb{N}. \tag{*}$$

It can be shown that $\phi_i(v_1,\ldots,v_{k_i}) \xrightarrow{*}_\Sigma w_i^m$ and that $t \xrightarrow{*}_\Sigma t^m = t\{w_1^m/\phi_1,\ldots, w_n^m/\phi_n\}$ for all $m$. Since $\theta(t)=t\{t_1/\phi_1,\ldots,t_n/\phi_n\}$ we have by Lemma 3.5.1 and (*) above,

$$\delta(\theta(t), t^m) \geq m.$$

One defines similarly $t'^m$ and one has

$$\delta(\theta(t'), t'^m) \geq m.$$

Since $\theta(t)=\theta(t')$ one has

$$\delta(t^m, t'^m) \geq m.$$

Hence, we have found $u=t^m$ and $u'=t'^m$ such that $t \xleftrightarrow{*}_\Sigma u$, $t' \xleftrightarrow{*}_\Sigma u'$ and $\delta(u,u')\geq m$.

Let us now prove the converse.

Let $t$ and $t'$, $u^m$ and $u'^m$ be such that $t \xleftrightarrow{*}_\Sigma u^m$, $t' \xleftrightarrow{*}_\Sigma u'^m$, $\delta(u^m, u'^m)\geq m$ for all $m$.

We can restrict $F$ and $V$ to the finite number of symbols appearing in $\Sigma$, $t$ and $t'$ so that $M^\infty(F\cup\Phi, V)$ is compact; hence one can find $u$ and $u'$ in $M^\infty(F\cup\Phi, V)$ and an increasing sequence $m_1<m_2<\cdots<m_k<\cdots$ such that

$$\mathop{\mathbf{Lim}}_{k\to\infty}(u^{m_k})=u, \qquad \mathop{\mathbf{Lim}}_{k\to\infty}(u'^{m_k})=u'.$$

Since $\Sigma$ is in Greibach normal form, $\theta$ is continuous and since $\theta(t)=\theta(u^m)$, $\theta(t')=\theta(u'^m)$,

$$\theta(u)=\mathop{\mathbf{Lim}}_{k\to\infty}\theta(u^{m_k})=\theta(t), \qquad \theta(u')=\mathop{\mathbf{Lim}}_{k\to\infty}\theta(u'^{m_k})=\theta(t').$$

Since $\theta$ is contracting,

$$d(\theta(u^{m_k}), \theta(u'^{m_k})) < d(u^{m_k}, u'^{m_k}) \leq (1/2)^{m_k},$$

hence

$$d(\theta(u), \theta(u'))=0,\ \theta(u)=\theta(u')$$

and finally

$$\theta(t)=\theta(t'), \quad \text{i.e. } t\equiv_\Sigma t'. \quad \square$$

## *5.8. The equality problem for algebraic trees: decidable subcases*

We prove a metatheorem on decidable cases and state (without proof) several applications.

**Theorem 5.8.1.** *If $\Sigma$ is an algebraic system in Greibach normal form such that $\equiv_\Sigma$ is finitely generated then $\equiv_\Sigma$ is decidable, even if one does not know the finite relation generating $\equiv_\Sigma$.*

The proof will use a technical definition. A binary relation $R \subseteq M(F \cup \Phi, V)^2$ is *self-proving* if for all $(s, s')$ in $R$:

(1) $\bar{\gamma}^*(\varepsilon, s) = \bar{\gamma}^*(\varepsilon, s') \in F_l$ for some $l > 0$,

(2) $\gamma(i, s) \overset{*}{\leftrightarrow}_R \gamma(i, s')$ for all $i = 1, \ldots, l$,

where $\gamma$ and $\bar{\gamma}^*$ are the mappings associated with $\Sigma$, defined in Section 5.6.

Let us denote by $\delta\theta$ the mapping

$$M(F \cup \Phi, V_k)^2 \to \mathbf{N} \cup \{\infty\}$$

defined by

$$\delta\theta(s, s') = \delta(\theta(s), \theta(s')).$$

**Lemma 5.8.2.** *If $R$ is such that $\delta\theta(s, s') \geq m$ for all $(s, s')$ in $R$ then the same holds for all $(s, s')$ such that $s \overset{*}{\leftrightarrow}_R s'$.*

**Proof.** By an induction the basic cases are as follows: if $\delta\theta(s_i, s_i') \geq m$ for all $i = 1, \ldots, k$ then

$$\delta\theta(s, s') \geq m \text{ if}$$

either $s = f(s_1, \ldots, s_k)$, $s' = f(s_1', \ldots, s_k')$, $f \in F_k$,

or $s = \phi(s_1, \ldots, s_k)$, $s' = \phi(s_1', \ldots, s_k')$, $\phi \in \Phi_k$

or $s = u[s_1, \ldots, s_k]$, $s' = u'[s_1, \ldots, s_k]$, $(u, u') \in R$. □

**Lemma 5.8.3.** *If $R$ is self-proving then $R$ is true, i.e. $R \subseteq \equiv_\Sigma$.*

**Proof.** Let us assume that $R$ is not contained in $\equiv_\Sigma$. Let $(t, t')$ be a pair such that $t \overset{*}{\leftrightarrow}_R t'$, $\delta\theta(t, t') < \infty$ and $\delta\theta(t, t') = \delta_0$ is minimal.

Lemma 5.8.2 shows that there exists $(s, s')$ in $R$ such that $\delta\theta(s, s') \leq \delta_0$. If $\delta_0 = 0$ then $\bar{\gamma}^*(\varepsilon, s) \neq \bar{\gamma}^*(\varepsilon, s')$ which contradicts the hypothesis that $R$ is self-proving. Hence $\delta_0 > 0$. We have $\theta(s) = f(\theta(s_1), \ldots, \theta(s_l))$, $f \in F_l$ with $s_i = \gamma(i, s)$ for all $i = 1, \ldots, l$, and similarly for $s'$. Hence $\delta\theta(\gamma(i, s), \gamma(i, s')) = \delta_0 - 1$ for some $i$ in $[l]$. Since $\gamma(i, s) \overset{*}{\leftrightarrow}_R \gamma(i, s')$ ($R$ being self-proving), this contradicts the choice of $\delta_0$. Hence $\delta\theta(t, t') = \infty$ for all $t, t'$ such that $t \overset{*}{\leftrightarrow}_R t'$ and in particular for all $(t, t')$ in $R$. This shows that $R \subseteq \equiv_\Sigma$. □

**Proof of Theorem 5.8.1.** Let us first remark that for a finite relation $R \subseteq M(F \cup \Phi, V)^2$, the property "$R$ is self-proving" is semi-decidable. This follows immediately from the definition.

Lemma 5.8.3 shows that $t \equiv_\Sigma t'$ if $t \overset{*}{\leftrightarrow}_R t'$ for some finite self-proving relation $R \subseteq M(F \cup \Phi, V)^2$. This sufficient condition is clearly semi-decidable.

Hence we only need to prove its necessity to achieve the proof.

To do so, we shall use (for the first time) the hypothesis that $\equiv_\Sigma$ is finitely generated.

Let $R_0$ be such that $\equiv_\Sigma$ is $\overset{*}{\leftrightarrow}_{R_0}$. If any component of any $(s, s')$ in $R_0$ is a variable say $v_i$ then the other must also be $v_i$ since we have assumed that $\phi(v_1, \ldots, v_k) \neq v_i$ for all $\phi$ in $\Phi$. We can clearly assume that $R_0$ does not contain any trivial pair such that $(v_i, v_i)$.

More generally we also exclude all pairs of the form $(t, t)$.

It follows that $R_0 \subseteq (M(F \cup \Phi, V) - (F_0 \cup V))^2$.

Let us now verify that $R_0$ is self-proving.

For all $(s, s')$ in $R_0$, $\theta(s) = \theta(s') \in M^\infty(F, V) - (F_0 \cup V)$ hence $\bar{\gamma}^*(\varepsilon, s) = \bar{\gamma}^*(\varepsilon, s') = f$ for some $f$ in $F_l$, with $l \geq 1$; we have $\theta(s) = f(\theta(\gamma(1, s)), \ldots, \theta(\gamma(l, s)))$ and similarly for $s'$. Hence $\gamma(i, s) \equiv_\Sigma \gamma(i, s')$ for all $i = 1, \ldots, l$ and $\gamma(i, s) \overset{*}{\leftrightarrow}_{R_0} \gamma(i, s')$ since $R_0$ generates $\equiv_\Sigma$. This establishes (2) and proves that $R_0$ is self-proving.

Hence if $t \equiv_\Sigma t'$ then $t \overset{*}{\leftrightarrow}_R t'$ for some finite self-proving $R$.

Hence $\equiv_\Sigma$ is semi-decidable; it is decidable by Theorem 5.7.1(3). □

In the following extension of Theorem 5.8.1 we shall be interested in deciding whether $t \equiv_\Sigma t'$ for $(t, t')$ in a recursive subset $\mathscr{C}$ of $M(F \cup \Phi, V)^2$.

Such a subset will be said *stable by OI-derivations* if for all $(s, s')$ in $\mathscr{C}$, $\bar{\gamma}^*(\varepsilon, s) = \bar{\gamma}^*(\varepsilon, s')$ is some $f$ in $F$ and for all $i = 1, \ldots, \rho(f)$, $(\gamma(i, s), \gamma(i, s')) \in \mathscr{C}$.

**Theorem 5.8.4.** *Let $\Sigma$ be in Greibach normal form, let $\mathscr{C} \subseteq M(F \cup \Phi, V)^2$ be recursive and stable by OI-derivations. Let us assume that there exists a finite subset $R_0$ of $\mathscr{C}$ such that for all $(t, t')$ in $\mathscr{C}$, $t \equiv_\Sigma t'$ if and only if $t \overset{*}{\leftrightarrow}_{R_0} t'$. Then $t \equiv_\Sigma t'$ is decidable for $(t, t')$ in $\mathscr{C}$.*

**Proof.** One can show that for $(t, t')$ in $\mathscr{C}$, $t \equiv_\Sigma t'$ if and only if there exists a finite self-proving relation $R \subseteq \mathscr{C}$ such that $t \overset{*}{\leftrightarrow}_R t'$.

The if part is exactly as in Theorem 5.8.1. The stability of $\mathscr{C}$ implies that $R_0$ minus the trivial pairs of the form $(t, t)$ is self-proving. Hence the only if part holds as well. □

Applications can be given to special classes of algebraic systems.

*5.8.5. Non-nested systems*

An algebraic system $\Sigma$ is *non-nested* if all the right-hand sides of its equations are *non-nested*, i.e. have no subterm of the form $\phi(s_1, \ldots, s_k)$ where some $s_j$ has an occurrence of a symbol in $\Phi$. One can put $\Sigma$ in such a form $\langle \phi_i(v_1, \ldots, v_{k_i}) = u_i; 1 \leq i \leq n \rangle$ that

(1) $u_i \in N = M(F, V \cup \Phi(M(F, V)))$ for all $i = 1, \ldots, n$,

(2) the components of the unique solution of $\Sigma$ are not finite,
(3) $\Sigma$ is trim.

Let now $\mathscr{C}$ be $N^2$ where $N$ is as above (i.e. is the set of trees having no more than one occurrence of a symbol in $\Phi$ on each branch). It can be shown that Theorem 5.8.4 is applicable with help of $R_0$ defined as follows:

$(s, s') \in R_0$ if and only if

$s = u[\tau_1, \ldots, \tau_k]$,

$s' = u'[\tau_{k+1}, \ldots, \tau_l]$,

$u = \phi_i(v_1, v_2, \ldots, v_k)$ for some $i$, and $k = k_i$,

$u' \in M(F, \Phi(V') \cup V')$ where $V' = \{v_{k+1}, \ldots, v_l\}$

and $u'$ has exactly one occurrence of each variable in $V'$, the substitution $\tau : V_l \to M(F, W)$ such that $\tau(v_i) = \tau_i$ is the most general unifier of the pair $(\theta(u), \theta(u'))$ (note that $\theta(u)$ and $\theta(u')$ are infinite trees).

A similar technique has been used in Courcelle and Franchi [21]. We conjecture that Theorem 5.8.1 is applicable to non-nested systems, giving the decidability of $\equiv_\Sigma$. This latter result is known to hold as a consequence of Valiant's result concerning finite turn DPDA's [68, 15].

*5.8.6. Monadic systems*

Let $\Sigma$ be a system in Greibach normal form, which is *monadic*, i.e. such that $\rho(\phi) \leq 1$ for all $\phi$ in $\Phi$.

It can be shown that $\equiv_\Sigma$ is finitely generated (Courcelle [20]) hence decidable by Theorem 5.8.1.

This decidability result answers an open question of Courcelle and Vuillemin [27]. It can also be obtained as a corollary of the decidability of the equivalence problem for stateless DPDA's [58] via Gallier's construction ([37] and Theorem 5.5.1).

*5.8.7. Yet another case*

Let $\Sigma$ be a trim algebraic system such that $\rho(\phi) \geq 1$ for all $\phi$ in $\Phi$. Let $\mathscr{C}$ be the set of pairs $(t, t')$ where $t \in M(\Phi_1, V_1)$ and $t' \in \Phi(M(\Phi_1, V_1))$ (where $\Phi_1 = \{\phi \in \Phi \mid \rho(\phi) = 1\}$).

It can be shown that the conditions of Theorem 5.8.4 are satisfied. The decidability result thus obtained is equivalent to the one of Harrison et al. [46] via Courcelle's construction ([15] and Theorem 5.5.1). The set $R_0$ of Theorem 5.8.4 is deduced via this construction of the set $\mathscr{C}$ of [19, (4.6)].

*5.9. First order unification of algebraic trees*

A natural object to investigate is $\mathbf{Unif}^{\infty}(t, t')$ for algebraic trees $t$ and $t'$. Such an investigation has already proved useful in Section 5.8.5.

A substitution $\tau : V_k \to M^\infty(F, W)$ is *algebraic* if $\tau(v_i)$ is algebraic for all $i \in [k]$.

The following theorem is fully similar to Theorem 4.15 except for the decidability, but this is not surprising.

**Theorem 5.9.1.** *Let $t$ and $t'$ be algebraic trees.*

(1) *If $t$ and $t'$ are unifiable, their most general unifier is algebraic.*

(2) *The equality problem for algebraic trees reduces to deciding whether* $\mathbf{Unif}^\infty(t, t') = \emptyset$ *or to deciding whether* $\mathbf{Unif}(t, t') = \emptyset$ *for algebraic trees $t$ and $t'$.*

**Proof.** (1) Let $t$ and $t'$ be unifiable. The proof of Theorem 4.15 gives a generalized regular system $S$ the right-hand side of each equation of which is a subtree of one of $t$ or $t'$, hence is algebraic. Hence Corollary 5.2.2 shows that the unique solution of $S$ is a tuple of algebraic trees.

Hence the most general unifier of $t$ and $t'$ is algebraic.

(2) It follows from the remark that $A(F, V) = A(F \cup V)$ that the equality problem for algebraic trees reduces to deciding whether $u \equiv_\Sigma u'$ for $u, u'$ in $M(F \cup \Phi)$. And $u \equiv_\Sigma u'$ is equivalent to $\mathbf{Unif}^\infty(\theta(u), \theta(u')) \neq \emptyset$ or to $\mathbf{Unif}(\theta(u), \theta(u')) \neq \emptyset$. □

## *5.10. Infinite trees and infinite words*

If all the symbols of $F$ are of arity 0 or 1 an element of $M^\infty(F)$ reduces to a finite or infinite word. More precisely, $M^\infty(F) = F_1^* F_0 \cup F_1^\omega$.

Let us define a word $w$ in $F_1^\omega$ (our notations concerning infinite words are borrowed from Nivat [54]) as *ultimately periodic* if $w = w_1 w_2^\omega$ for some $w_1$ in $F_1^*$ and some $w_2$ in $F_1^+$. Let $\mathbf{Ult}(F_1)$ be the set of such words. Then we have the following result.

**Proposition 5.10.1.** $R(F) = A(F) = F_1^* F_0 \cup \mathbf{Ult}(F_1)$.

**Proof.** It is easy to prove that $R(F) = F_1^* F_0 \cup \mathbf{Ult}(F_1)$. Since $R(F) \subseteq A(F)$, we only have to prove that $A(F) \subseteq F_1^* F_0 \cup \mathbf{Ult}(F_1)$.

Let $\Sigma$ be an algebraic system over $F$. It may be assumed trim. As usual, we take it of the form $\Sigma = \langle \phi_1(v_1, \ldots, v_{k_1}) = u_1, \ldots, \phi_n(v_1, \ldots, v_{k_n}) = u_n \rangle$ and let $(t_1, \ldots, t_n)$ be its solution.

For each $i = 1, \ldots, n$,

(1) either $t_i \in M^\infty(F)$ and then $k_i = 0$,

(2) or $t_i \in M^\infty(F, V_{k_i}) - M^\infty(F)$ and, since $F = F_1 \cup F_0$, $t_i$ has exactly one occurrence of a variable, hence $k_i = 1$ and $t_i \in M(F_1, \{v_1\}) = F_1^* v_1$.

We can assume that case (1) holds for $1 \leq i \leq l$ and case (2) for $l + 1 \leq i \leq n$.

One can translate $\Sigma$ into $(\Sigma', \phi_1, \ldots, \phi_l, t_{l+1}, \ldots, t_n)$ where $\Sigma' = \langle \phi_1 = u'_1, \ldots, \phi_l = u'_l \rangle$ and $u'_i = u_i\{t_{l+1}/\phi_{l+1}, \ldots, t_n/\phi_n\}$ for $i = l + 1, \ldots, n$. Since all unknowns of $\Sigma'$ are of arity 0, $\Sigma'$ is regular. It follows that $t_1, \ldots, t_n$ are in $F_1^* F_0 \cup \mathbf{Ult}(F_1)$ hence $A_\Sigma(F) = A_{\Sigma'}(F) \subseteq F_1^* F_0 \cup \mathbf{Ult}(F_1)$. □

**Remark 5.10.2.** This proposition shows that the concept of *algebraic tree* has no counterpart in infinite words, whereas regular infinite trees correspond to ultimately periodic infinite words.

## 6. Conclusion

The present work has studied several aspects of finite and infinite trees which are especially relevant to the theory of computing. To summarize:

(1) A double theory of infinite trees, by topological or order-theoretical methods has been developed.

(2) First-order and second-order substitutions are two important concepts; some of their basic combinatorial properties have been stated; their continuity properties have been investigated in detail.

(3) Regular trees and their relations with first-order unification have been studied; rational expressions denoting regular trees have been introduced and related with iterative theory expressions.

(4) Algebraic trees have been studied; their combinatorial properties are complex enough to yield an open problem which is interreducible with the equivalence problem for DPDA's; decidable special cases have been stated.

Many other interesting aspects (raising open problems) could have been treated as well (except for the author's time availability):

(5) Higher-order algebraic trees corresponding to higher-order recursive program schemes (Damm et al. [30, 31], Gallier [39]), algebraic trees as images under yield operators of regular trees,

(6) Frontiers of infinite trees as generalized infinite words (Courcelle [16], Heilbrunner [47]) and even more important:

(7) Extensions of congruences from finite trees to infinite ones: whereas the theory is rather neat in the approach with partial orders, (Courcelle [18]), it is much more difficult in the topological approach (Courcelle [17]).

And finally, the *theory of tree languages* which constitutes a theory by itself but is of course grounded on the present the *theory of infinite trees*.

## Acknowledgment

This paper is dedicated to the memory of the late C.C. Elgot whose contribution to the theory of infinite trees is essential.

I thank S. Bloom, G. Cousineau, G. Huet, R. Tindell and some of the participants of the Marktoberdorf summer school for their comments which have been useful for the preparation of the final version of the paper.

And last but not least, I thank Mrs Polzin for (two) beautiful typings.

## References


[1] A. Arnold and M. Dauchet, Théorie des magmoïdes, *RAIRO Inform. Théor.* **12** (1978) 235–257.
[2] A. Arnold and M. Nivat, Metric interpretations of infinite trees and semantics of non deterministic recursive programs, *Theoret. Comput. Sci.* **11** (1980) 181–205.
[3] A. Arnold and M. Nivat, The metric space of infinite trees. Algebraic and topological properties, *Fund. Inform.* **III.4** (1980) 445–476.
[4] H. Bekič, Definable operations in general algebras, and the theory of automata and flowcharts, Unpublished work, IBM Laboratory, Vienna (1969).
[5] S. Bloom, All solutions of a system of recursion equations in infinite trees and other contraction theories, *J. Comput. System Sci.*, to appear.
[6] S. Bloom and C. Elgot, The existence and construction of free iterative theories, *J. Comput. System Sci.* **12** (1976) 305–318.
[7] S. Bloom, C. Elgot and J. Wright, Solutions of the iteration equation and extensions of the scalar iteration operation, *SIAM J. Comput.* **9** (1980) 25–45.
[8] S. Bloom, S. Ginali and J. Rutledge, Scalar and vector iteration, *J. Comput. System Sci.* **14** (1977) 251–256.
[9] S. Bloom and D. Patterson, Easy solutions are hard to find, *Proc. 6th Colloquium on Trees in Algebra and Programming*, Lecture Notes in Computer Science **112** (Springer, Berlin, 1981) 135–146.
[10] S. Bloom and R. Tindell, Compatible orderings on the metric theory of trees, *SIAM J. Comput.* **9** (1980) 683–691.
[11] S. Bloom and R. Tindell, Varieties of 'if-then-else', Submitted for publication (1981).
[12] R. Book, The undecidability of a word problem: on a conjecture of Strong, Maggiolo-Schettini and Rosen, *Inform. Process Lett.* **12** (1981) 121–122.
[13] P. Casteran, Structures de contrôle: définitions opérationnelles et algébriques, Thèse de 3ème cycle, University Paris-7 (1979).
[14] B. Courcelle, On jump-deterministic pushdown automata, *Math. Systems Theory* **11** (1977) 87–109.
[15] B. Courcelle, A representation of trees by languages, *Theoret. Comput. Sci.* **6** (1978) 255–279 and **7** (1978) 25–55.
[16] B. Courcelle, Frontiers of infinite trees, *RAIRO Inform. Théor.* **12** (1978) 319–337.
[17] B. Courcelle, Arbres infinis et systèmes d'équations, *RAIRO Inform. Théor.* **13** (1979) 31–48.
[18] B. Courcelle, Infinite trees in normal form and recursive equations having a unique solution, *Math. Systems Theory* **13** (1979) 131–180.
[19] B. Courcelle, An axiomatic approach to the Korenjak–Hopcroft algorithms, *Math. Systems Theory*, to appear.
[20] B. Courcelle, Work in preparation.
[21] B. Courcelle and P. Franchi-Zannettacci, On the equivalence problem for attribute systems, *Information and Control*, to appear.
[22] B. Courcelle and I. Guessarian, On some classes of interpretations, *J. Comput. System Sci.* **17** (1978) 388–413.
[23] B. Courcelle, G. Kahn and J. Vuillemin, Algorithmes d'équivalence et de réduction à des expressions minimales, dans une classe d'équations récursives simples, *Proc. 2nd International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science **14** (Springer, Berlin, 1974) 200–213.
[24] B. Courcelle and M. Nivat, Algebraic families of interpretations, *Proc. Annual Symposium on Foundations of Computer Science*, Houston, TX (1976) 137–146.
[25] B. Courcelle and M. Nivat, The algebraic semantics of recursive program schemes, in: *Mathematical Foundations of Computer Science '78*, Lecture Notes in Computer Science **64** (Springer, Berlin, 1978) 16–30.
[26] B. Courcelle and J.C. Raoult, Completions of ordered magmas, *Fund. Inform.* **III.1** (1980) 105–116.
[27] B. Courcelle and J. Vuillemin, Completeness results for the equivalence of recursive schemes, *J. Comput. System Sci.* **12** (1976) 179–197.
[28] G. Cousineau, La programmation en EXEL, *Rev. Tech. Thomson-CSF* **10** (1978) 209–234 and **11** (1979) 13–35.

[29] G. Cousineau, An algebraic definition for control structures, *Theoret. Comput. Sci.* **12** (1980) 175–192.
[30] W. Damm, The IO- and OI-hierarchies, *Theoret. Comput. Sci.* **20** (1982) 95–207.
[31] W. Damm, E. Fehr and K. Indermark, Higher type recursion and self-application as control structures, in: E. Neuhold, Ed., *Formal Descriptions of Programming Concepts* (North-Holland, Amsterdam, 1978) 461–487.
[32] C. Elgot, Monadic computation and iterative algebraic theories, in: H.E. Rose, Ed., *Logic Colloquium* **73** (North-Holland, Amsterdam, 1975) 175–230.
[33] C. Elgot, Structured programming with and without GOTO statements, *IEEE Trans. Software Engrg.* **2** (1976) 41–54.
[34] C. Elgot, S. Bloom and R. Tindell, The algebraic structure of rooted trees, *J. Comput. System Sci.* **16** (1978) 362–399.
[35] J. Engelfriet and E. Schmidt, IO and OI, *J. Comput. System Sci.* **15** (1977) 328–361 and **16** (1978) 67–99.
[36] P. Enjalbert, Systèmes de déductions pour les arbres et les schémas de programmes, *RAIRO Inform. Théor.* **14** (1980) 247–278 and **15** (1981) 3–21.
[37] J. Gallier, DPDA's in 'atomic normal form' and applications to the equivalence problems, *Theoret. Comput. Sci.* **14** (1981) 155–186.
[38] J. Gallier, Recursion closed algebraic theories, *J. Comput. System Sci.*, to appear.
[39] J. Gallier, *N*-rational algebras, I: Basic Properties and free algebras, II: Varieties and the logic of inequalities, Submitted for publication.
[40] S. Ginali, Regular trees and the free iterative theory, *J. Comput. System Sci.* **18** (1979) 228–242.
[41] J. Goguen, J. Thatcher, E. Wagner and J. Wright, Initial algebra semantics and continuous algebras, *J. ACM* **24** (1977) 68–95.
[42] S. Gorn, Explicit definitions and linguistic dominoes, in: J. Hart and S. Takasu, Eds., *Systems and Computer Science* (University of Toronto Press, 1967) 77–105.
[43] I. Guessarian, Program transformations and algebraic semantics, *Theoret. Comput. Sci.* **9** (1979) 39–65.
[44] I. Guessarian, *Algebraic Semantics*, Lecture Notes in Computer Science **99** (Springer, Berlin, 1981).
[45] M. Harrison, *Introduction to Formal Language Theory* (Addison-Wesley, Reading, MA, 1978).
[46] M. Harrison, I. Havel and A. Yehudai, On equivalence of grammars through transformation trees, *Theoret. Comput. Sci.* **9** (1979) 173–205.
[47] S. Heilbrunner, An algorithm for the solution of fixed-point equations for infinite words, *RAIRO Inform. Théor.* **13** (1979) 131–141.
[48] G. Huet, Résolution d'équations dans les langages d'ordre 1, 2, ... ω, Doctoral dissertation, University Paris-7 (1976).
[49] G. Huet, Confluent reductions: abstract properties and applications to term rewriting systems, *J. ACM* **27** (1980) 797–821.
[50] J. Leszczylowski, A theorem on resolving equations in the space of languages, *Bull. Acad. Polon. Sci., Ser. Sci. Math. Astronom. Phys.* **19** (1979) 967–970.
[51] G. Markowsky and B. Rosen, Bases for chain-complete posets, *IBM J. Res. Develop.* **20** (1976) 138–147.
[52] J. Mycielski and W. Taylor, A compactification of the algebra of terms, *Algebra Universalis* **6** (1976) 159–163.
[53] M. Nivat, On the interpretation of recursive polyadic program schemes, Symposia Mathematica **15** (Academic Press, New York, 1975) 255–281.
[54] M. Nivat, Mots infinis engendrés par une grammaire algébrique, *RAIRO Inform. Théor.* **11** (1977) 311–327.
[55] M. Nivat, Private communication.
[56] L. Nolin and G. Ruggiu, A formalization of EXEL, *Proc. ACM Symposium on Principles of Programming Languages*, Boston (1973).
[57] M. O'Donnell, *Computing in Systems Described by Equations*, Lecture Notes in Computer Science **58** (Springer, Berlin, 1977).
[58] M. Oyamaguchi and N. Honda, The decidability of the equivalence for deterministic stateless pushdown automata, *Information and Control* **38** (1978) 367–376.

[59] M. Oyamaguchi, N. Honda and Y. Inagaki, The equivalence problem for real-time strict deterministic languages, *Information and Control* **45** (1980) 90–115.
[60] M. Paterson and M. Wegman, Linear unification, *J. Comput. System Sci.* **16** (1978) 158–167.
[61] J. Robinson, A machine-oriented logic based on the resolution principle, *J. ACM* **12** (1965) 23–41.
[62] B. Rosen, Tree-manipulating systems and Church–Rosser theorems, *J. ACM* **20** (1973) 160–187.
[63] B. Rosen, Program equivalence and context-free grammars, *J. Comput. System Sci.* **11** (1975) 358–374.
[64] M. Schützenberger, On context-free languages and push-down automata, *Information and Control* **6** (1963) 246–264.
[65] J. Tiuryn, Fixed points and algebras with infinitely long expressions, *Fund. Inform.* **II** (1978) 107–128 and **II** (1979) 317–335.
[66] J. Tiuryn, On a connection between regular algebras and rational algebraic theories, *Proc. 2nd Workshop on Categorical and Algebraic Methods in Computer Science and System Theory*, Dortmund, West Germany (1978).
[67] J. Tiuryn, Unique fixed points vs. least fixed points, Report 49, RWTH Aachen, West Germany (1978).
[68] L. Valiant, The equivalence problem for deterministic finite-turn push-down automata, *Information and Control* **25** (1974) 123–133.
[69] J. Wright, J. Thatcher, E. Wagner and J. Goguen, Rational algebraic theories and fixed-point solutions, *Proc. 17th Symposium on Foundations of Computer Science*, Houston, TX (1976) 147–158.
[70] J. Wright, E. Wagner and J. Thatcher, A uniform approach to inductive posets and inductive closure, *Theoret. Comput. Sci.* **7** (1978) 57–77.

## Note added in proof

L. Boasson has shown that the answer to problem 5.5.2 is negative.