On Infinite Terms Having a Decidable Monadic Theory

Didier Caucal

IRISA–CNRS, Campus de Beaulieu, 35042 Rennes, France,
caucal@irisa.fr

Abstract. We study a transformation on terms consisting of applying an inverse deterministic rational mapping followed by an unfolding. Iterating these transformations from the regular terms gives a hierarchy of families of terms having a decidable monadic theory. In particular, the family at level 2 contains the morphic infinite words investigated by Carton and Thomas. We show that this hierarchy coincides with the hierarchy considered by Knapik, Niwiński and Urzyczyn: the families of terms that are solutions of higher order safe schemes. We also show that this hierarchy coincides with the hierarchy defined by Damm, and recently considered by Courcelle and Knapik: the families of terms obtained by iterating applications of first order substitutions to the set of regular terms. Finally, using second order substitutions yields the same terms.

1 Introduction

A general approach to check properties of a finite system is to express these properties by formulas, that can be decided on the generally infinite structure modeling the behaviour of the system. We focus on monadic second order formulas and the systems we consider are the higher order schemes [In76], [Da82] and the deterministic pushdown automata with multi-level stacks [En83].

A simple way to show the decidability of the monadic second order theory of a given graph is to obtain the graph from a finite graph using basic graph transformations that preserve the decidability of the monadic theory. We only use two graph transformations. The first transformation is the unfolding of a graph from a given vertex [CW98]. For instance, the unfoldings of finite graphs are the regular trees (having only a finite number of non isomorphic subtrees) whose each node is of finite out-degree. Therefore, these trees have a decidable monadic theory; it is the case in particular for the complete infinite binary tree [Ra69]. The second graph transformation is given by a finite automaton over labels [Ca96]. Precisely, a rational mapping h associates to each label a a rational language $h(a)$ (over labels and barred labels for moving by inverse arc). And we apply h^{-1} to a graph G to get the graph $h^{-1}(G)$ having an arc $s \xrightarrow{a} t$ when there is a path $s \overset{u}{\Longrightarrow} t$ in G for some word u in $h(a)$. This graph transformation preserves the decidability of the monadic theory: it is a noncopying monadic

K. Diks et al. (Eds): MFSC 2002, LNCS 2420, pp. 165–176, 2002.

second-order definable transduction in the sense of [Co94]. This transformation has a maximality property: starting from the regular trees, we get the same graphs that we can get by monadic definable transductions [Ba98] : the prefix-recognizable graphs [Ca96].

By alternate repetition of unfoldings and inverse rational mappings from the finite trees, we get a hierarchy of tree families and a hierarchy of graph families that have a decidable monadic theory. At level 0, the tree family is by definition the set of finite trees, and the graph family is the set of finite graphs. At level 1, the tree family is the set of regular trees, and the graph family is the set of prefix-recognizable graphs. At higher levels, the tree family and the graph family have never been studied. However for two related hierarchies of (families of finite and infinite) terms, the decidability of the monadic theory has been shown. The first one classifies the solutions of higher order safe schemes [KNU02]. The second one is obtained by iterating first order substitutions starting from regular terms [CK02]. As terms are deterministic trees, we restrict our hierarchy of tree families by using only deterministic rational mappings from terms to terms, and starting from the regular terms. The family at level 2 contains all the morphic infinite words [CT00] (Proposition 3.2). Then, we show that the three hierarchies of term families are equal (Theorems 3.3 and 3.5). In particular, we describe paths in lambda graphs of [KNU02] by giving directly a deterministic rational mapping. And we show that the evaluation of first order substitutions in [CK02] is essentially an inverse deterministic rational mapping followed by an unfolding. Furthermore we show that the evaluation of second order substitutions can be done by applying two inverse deterministic rational mappings each being followed by an unfolding (Proposition 3.4).

2 A Hierarchy of Tree Families

We present two basic graph transformations that preserve the decidability of the monadic theory: the unfolding and the inverse rational mapping. Iterating these transformations gives a hierarchy of graph families, and especially a hierarchy of tree families.

Let $\mathbb{N}$ be the set of nonnegative integers. For any $n \in \mathbb{N}$, we denote $[n] = \{1,\ldots,n\}$ with $[0] = \emptyset$. For any set E, we denote $|E|$ its cardinal and 2^E its powerset. Let L^* be the free monoid generated by any set L of symbols, called *letters*. Any *word* $u \in L^*$ of *length* $|u| \in \mathbb{N}$ is a mapping from $\{1,\ldots,|u|\}$ into L represented by $u(1)\ldots u(|u|) = u$. The word of length 0 is the empty word ε. For any word u, $Occ(u) := \{\ u(i) \mid i \in [|u|]\ \}$ is its set of letters.

Let L be a countable set of symbols for labelling arcs. A simple, oriented and arc labelled *graph* G is a subset of $V{\times}L{\times}V$ where V is an arbitrary set and such that its *label* set

$$L_G := \{\ a \in L \mid \exists\ s,t,\ (s,a,t) \in G\ \} \quad \text{is finite}$$

but its *vertex* set

$$V_G := \{\ s \mid \exists\ a,t,\ (s,a,t) \in G\ \vee\ (t,a,s) \in G\ \} \quad \text{is finite or countable.}$$

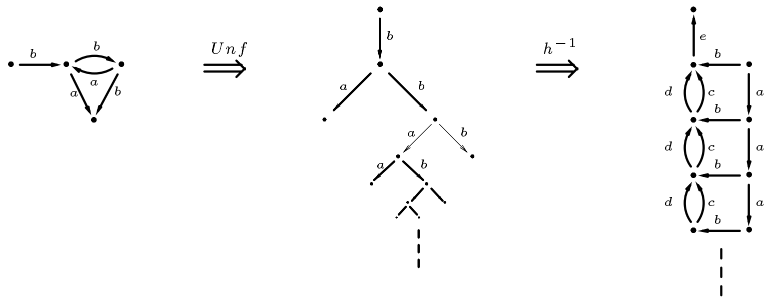

Fig. 2.2. A root unfolding and an inverse finite mapping.

Any (s,a,t) of G is a *labelled arc* of *source* s, of *target* t, with label a, and is identified with the labelled transition $s \xrightarrow[G]{a} t$ or directly $s \xrightarrow{a} t$ if G is understood. For instance, the finite graph $\{r \xrightarrow{b} p\,,\, p \xrightarrow{a} s\,,\, p \xrightarrow{b} q\,,\, q \xrightarrow{a} p\,,\, q \xrightarrow{b} s\}$ has p,q,r,s for vertices, and has a,b for labels, and is represented by the leftmost figure in Figure 2.2. A vertex s is *terminal* if it is source of no arc. We write $s \xrightarrow[G]{P}$ with $P \subseteq L$ if there is $s \xrightarrow[G]{a} t$ for some $a \in P$ and $t \in V_G$. We denote $G_{|W} \ := \ \{\ s \xrightarrow[G]{a} t \mid s,t \in W\ \}$ the *restriction* of a graph G to a subset W. Any word $s_0 a_1 s_1 \ldots a_n s_n$ such that $n \geq 0$ and $s_0 \xrightarrow{a_1} s_1 \ldots s_{n-1} \xrightarrow{a_n} s_n$ is a *path* from s_0 to s_n labelled by $w = a_1 \ldots a_n$, and we write $s_0 \overset{w}{\underset{G}{\Longrightarrow}} s_n$ or directly $s_0 \overset{w}{\Longrightarrow} s_n$ if G is understood. We say that a vertex r is a *root* of G if every vertex is accessible from r: $\forall\ s \in V_G,\ \exists\ w\,,\ r \overset{w}{\Longrightarrow} s$. We denote $Path(G)$ the path language of G, and $L(G,I,F)$ the path labels from a set I to a set F:

$$L(G,I,F) \ := \ \{\ w \mid \exists\ s \in I,\ \exists\ t \in F,\ s \overset{w}{\underset{G}{\Longrightarrow}} t\ \}$$

For instance taking the leftmost graph in Figure 2.2, its path labels from its root to its terminal vertex is $b(ba)^*(a+bb)$. A *trace* of a graph is a path language from and to finite vertex sets. Recall that the traces of finite graphs form the set

$$Rat(L^*) := \{\ L(G,I,F) \mid |G| < \infty\ \wedge\ I,F \subseteq V_G\ \}$$

of *rational languages* over L, and we denote $Fin(L^*)$ the family of finite languages over L. A graph is *deterministic* if distinct arcs with the same source have distinct labels: if $r \xrightarrow{a} s$ and $r \xrightarrow{a} t$ then $s = t$. Recall that a graph is a *tree* if it has a root r which is target of no arc, and every vertex $s \neq r$ is target of a unique arc. Any vertex of a tree is also called a *node* and any terminal node is a *leaf.* The *subtree* of a tree G at node s is the restriction $G_{|\{\,t \mid \exists\, w,\ s \overset{w}{\Longrightarrow} t\}}$ of G to the vertices accessible from s. A *forest* is a graph such that each connected component is a tree.

We will present two known basic graph transformations preserving the graph isomorphism, and also the decidability of the monadic theory. Let us recall the notion of graph isomorphism and the monadic theory of a graph.

Given a binary relation R, we consider the graph

$$R(G) \ := \ \{\ s' \xrightarrow{a} t' \mid \exists\ s \xrightarrow[G]{a} t,\ s\ R\ s'\ \wedge\ t\ R\ t'\ \}$$

of the *application* of R to G. We say that $R(G)$ is *isomorphic* to G when R is a bijection from V_G to $V_{R(G)}$, and we write $G \simeq R(G)$.

To construct monadic second-order formulas, we take two disjoint countable sets: a set of *vertex variables* and a set of *vertex set variables. Atomic formulas* have one of the following two forms:

$$x \in X \quad \text{or} \quad x \xrightarrow{a} y$$

where X is a vertex set variable, x and y are vertex variables, and $a \in L$.

From the atomic formulas, we construct as usual the *monadic second-order formulas* with the propositional connectives $\neg, \wedge$ and the existential quantifier $\exists$ acting on these two kinds of variables. A *sentence* is a formula without free variable. The set of monadic second-order sentences $MTh(G)$ satisfied by a graph G forms the *monadic theory* of G. Note that two isomorphic graphs satisfy the same sentences: $MTh(G) = MTh(H)$ if $G \simeq H$. Many articles concern the decidability of the monadic theory for infinite graphs (see among others [Sem84], [MuS85], [Co90], [Th90] and [Th97]).

We present now two known graph transformations: the unfolding and the inverse label mapping. The first transformation is to unfold a graph from all its vertices. The *unfolding* $Unf(G)$ of any graph G is the following forest:

$$Unf(G) \ := \ \{ \ ws \xrightarrow{a} wsat \mid wsat \in Path(G) \ \wedge \ s \underset{G}{\xrightarrow{a}} t \ \}$$

The unfolding $Unf(G,I)$ of a graph G from a vertex subset I is the restriction of $Unf(G)$ to paths starting from vertices in I *i.e.*

$$Unf(G,I) \ := \ Unf(G)_{|\{ \, su \in Path(G) \mid s \in I \, \}}$$

In particular $Unf(G) = Unf(G,V_G)$ and for any vertex $s \in V_G$, $Unf(G,s)$ is a tree (a connected component of $Unf(G)$), called an *unfolding tree* of G. For instance, the unfolding tree from the root of the leftmost graph in Figure 2.2 is given by the middle representation. This tree is deterministic and is a *regular tree*: it has a finite number of non isomorphic subtrees.

Note that for any deterministic graph G, its unfolding $Unf(G,r)$ from any vertex r is isomorphic to the following tree:

$$Tree(G,r) := \{ \ u \xrightarrow{a} ua \mid \exists \ s, \ r \underset{G}{\overset{ua}{\Longrightarrow}} s \ \wedge \ a \in L \ \}$$

where the isomorphism associates to any $s_0 a_1 s_1 \ldots a_n s_n \in Path(G)$ its label $a_1 \ldots a_n$. The unfolding preserves the decidability of the monadic theory.

Proposition 2.1 [CW98] *Given a graph G and a vertex s, we have*

$$MTh(G) \ \textit{is decidable} \ \Longrightarrow \ MTh(Unf(G,s)) \ \textit{is decidable.}$$

The second transformation is the inverse label mapping [Ca96]. To move by inverse arcs, we take a new symbol set $\overline{L} := \{ \ \overline{a} \mid a \in L \ \}$ in bijection with L. Any transition $s \underset{G}{\xrightarrow{\overline{a}}} t$ means that $t \underset{G}{\xrightarrow{a}} s$ is an arc of G. We extend the existence of a path $\overset{w}{\Longrightarrow}$ labelled by a word w in $(L \cup \overline{L})^*$: $s \overset{\varepsilon}{\Longrightarrow} s$ and $s \overset{aw}{\Longrightarrow} t$ if there is r such that $s \xrightarrow{a} r \overset{w}{\Longrightarrow} t$. Given any relation $h \subseteq L \times (L \cup \overline{L})^*$ of finite domain *i.e.* a mapping from L into $2^{(L \cup \overline{L})^*}$ such that $Dom(h) := \{ \ a \mid h(a) \neq \emptyset \ \}$ is finite, the *inverse mapping* $h^{-1}(G)$ of any graph G by h is the following graph:

$$h^{-1}(G) \ := \ \{ \ s \xrightarrow{a} t \mid \exists \ w \in h(a), \ s \underset{G}{\overset{w}{\Longrightarrow}} t \ \}$$

and h is a *finite mapping* (resp. *rational mapping*) when $h(a)$ is finite (resp.

rational) for every $a \in L$. For instance, starting from the tree of the middle representation of Figure 2.2 and by applying by inverse the finite mapping defined by $a \mapsto \{\overline{a}aba\}$; $b \mapsto \{\overline{a}aa\}$; $c\,,d \mapsto \{\overline{a}\,\overline{a}\overline{b}\overline{a}aa\}$; $e \mapsto \{\overline{a}\,\overline{a}\overline{b}\,\overline{b}ba\}$, we get the deterministic graph of the rightmost representation of Figure 2.2 . The inverse rational mapping preserves the decidability of the monadic theory.

Proposition 2.3 [Ca96] *Given a graph G and a rational mapping h ,*

$$MTh(G) \text{ is decidable } \Longrightarrow MTh(h^{-1}(G)) \text{ is decidable.}$$

Starting from the finite trees and by alternate repetition of inverse rational mappings and unfoldings, we get a hierarchy of families of trees and a hierarchy of families of graphs having a decidable monadic theory. Precisely and for any graph family $\mathcal{F}$, we denote

$[\mathcal{F}] := \{\ H \mid \exists\ G \in \mathcal{F},\ G \simeq H\ \}$ the closure by isomorphism of $\mathcal{F}$

$Unf(\mathcal{F}) := [\{\ Unf(G,s) \mid \exists\ G \in \mathcal{F}\ s \in V_G\ \}]$

the unfolding trees, up to isomorphism, of graphs in $\mathcal{F}$

$Rat^{-1}(\mathcal{F}) := \{\ h^{-1}(G) \mid G \in \mathcal{F}\ \wedge\ h : L \longrightarrow Rat((L \cup \overline{L})^*)\ \wedge\ |Dom(h)| < \infty\ \}$

the inverse rational mappings of graphs in $\mathcal{F}$.

Taking the set $Tree_0$ of finite trees, we consider the hierarchy $(Tree_n)_{n \geq 0}$ of trees and the hierarchy $(Graph_n)_{n \geq 0}$ of graphs, defined inductively on $n \geq 0$ as follows:

$$Graph_n := Rat^{-1}(Tree_n) \quad \text{and} \quad Tree_{n+1} := Unf(Graph_n)$$

By Proposition 2.1 and 2.3, $\bigcup_{n \geq 0} Graph_n$ is a family of graphs having a decidable monadic theory.

At level 0, by definition $Tree_0$ is the family of finite trees, and $Graph_0$ is the family of finite graphs whose their traces are the rational languages.

At level 1, $Tree_1$ is the family of regular trees of finite degree (each node is node of a finite number of edges). Let us describe the family $Graph_1$.

The family $Graph_1$ is the family of prefix-recognizable graphs [Ca96] : it is the set of prefix transition graphs of labelled recognizable word rewriting systems, or equivalently the set of VR-equational graphs [Ba98]. We get the graphs of $Graph_1$ by ε-closure of the transition graphs of pushdown automata [MuS85] [Ca90] with ε-transitions. An important property is that the inverse rational mappings applied to $Tree_1$ to get $Graph_1 = Rat^{-1}(Tree_1)$ are sufficient to obtain all the graphs which are monadic interpretable on $Tree_1$ (or only on the binary tree) [Ba98]. Note that $Graph_1$ trace the context-free languages, and the trees in $Graph_1$ are all the regular trees. An extension of $Graph_1$ to hypergraphs has been done in [LN01] and [Bl02].

At level $n \geq 2$, we have no general result of $Graph_n$. However we have results on hierarchies of families of deterministic trees, or more exactly of finite and infinite terms [Da82], [KNU01], [KNU02], [CK02], and we will compare these hierarchies with $(Tree_n)_{n\,\geq\,0}$ restricted to terms.

3 A Hierarchy of Term Families

We restrict the previous hierarchy on tree families to a hierarchy of term families $(Term_n)$ obtained from the family of regular terms by iterated inverse determin-

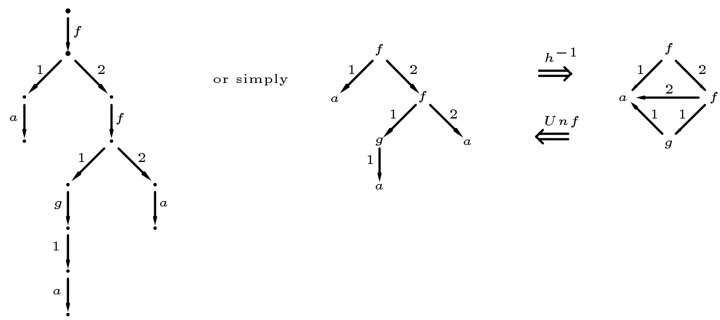

Fig. 3.1. Tree representations and an inverse finite mapping.

istic rational mappings with unfoldings. Any morphic infinite word is in $Term_2$ (Proposition 3.2). We establish that the hierarchy $(Term_n)$ coincides with the hierarchy (Sub_n) of families of terms obtained from the regular terms by iterated first order substitutions (Theorem 3.3). Then, we show that the terms obtained from the regular terms by iterated second order substitutions are the terms of the hierarchy (Proposition 3.4). Finally, we establish that the hierarchy $(Term_n)$ coincides with the hierarchy of families of terms that are solutions of safe schemes (Theorem 3.5).

Let F be a set of symbols called *functions*, graded by a mapping $\varrho : F \longrightarrow \mathbb{N}$ associating to each function f its *arity* $\varrho(f)$, and such that

$$F_n := \{\ f \in F \mid \varrho(f) = n\ \} \ \text{ is countable for every } n \geq 0.$$

The set $T(F)$ of *finite terms* is the smallest subset of F^* such that

$$f \in F\ \wedge\ t_1, \ldots, t_{\varrho(f)} \in T(F) \quad \Longrightarrow \quad f t_1 \ldots t_{\varrho(f)} \in T(F)$$

Particularly the *constant* set $F_0 \subseteq T(F)$. Any finite term, for instance $fafgaa$ with $\varrho(f) = 2\,, \varrho(g) = 1\,, \varrho(a) = 0$, is represented by a tree as shown by the leftmost representation of Figure 3.1 . The middle representation of Figure 3.1 is simpler and usual. So we have to use vertex labelled graphs.

A coloured graph is a graph with a vertex labelling in a finite subset of F. Precisely, a *coloured graph* $G := \underline{G} \cup c_G$ is the union of a (uncoloured) graph $\underline{G} \subseteq V{\times}L{\times}V$ and a *vertex labelling* or *colouring* $c_G \subseteq V{\times}F$ which is functional: $|c_G \cap \{s\}{\times}F| \leq 1$ for every $s \in V$, such that its domain $\{\, s \mid \exists\, f,\ (s,f) \in G\,\}$ is the *vertex set* V_G of G containing $V_{\underline{G}}$, and such that its image $\{\, f \mid \exists\, s,\ (s,f) \in G\,\}$ is finite: each vertex has one colour from a finite set. Any $(s,f) \in G$ is a vertex s labelled f and is also denoted sf. Note that the coloured graph $\{1a\}$ has an empty uncoloured graph and the vertex labelling $1 \mapsto a$.

Any path $s_0a_1s_1 \ldots a_ns_n \in Path(\underline{G})$ is now labelled by $w = c(s_0)a_1c(s_1) \ldots a_nc(s_n)$ and we write $s_0 \overset{w}{\underset{G}{\Longrightarrow}} s_n$ (or directly $s_0 \overset{w}{\Longrightarrow} s_n$ if G is understood) for a path from s_0 to s_n labelled by w. For instance, the path labels $L(G, r, E)$ of the coloured tree G of Figure 3.1 from its root r to its set E of its leaves is the language $\{f1a, f2f1g1a, f2f2a\}$ called the *branch language* of G [Co83].

We extend the unfolding and the inverse mapping to any coloured graph G:

$$Unf(G) := Unf(\underline{G}) \ \cup \ \{\ wsf \mid ws \in Path(\underline{G}) \ \wedge \ sf \in G\ \}$$

and $h^{-1}(G) := \{\ s \xrightarrow{a} t \mid \exists\, w \in h(a),\ s \underset{G}{\overset{w}{\Longrightarrow}} t\ \}$

$$\cup\ \{\ s\,h(f) \in G \mid \exists\, a\ \exists\, w \in h(a)\ \exists t,\ s \underset{G}{\overset{w}{\Longrightarrow}} t \ \vee\ t \underset{G}{\overset{w}{\Longrightarrow}} s\ \}$$

where h is a mapping from $L \cup F$ into $2^{(L \cup \overline{L} \cup F)^*}$ such that $h(F) \subseteq F$; so the vertices of $h^{-1}(G)$ are the vertices s of $\underline{h^{-1}(G)}$ coloured by $h(f)$ if s is coloured by f in G.

For any coloured graph G and any vertex r, we also consider the unfolding

$$Unf(G,r) := Unf(G)_{|\{\, rw \,|\, rw \in Path(\underline{G})\,\}}$$

of G from r, and a canonical representative for $\underline{G}$ deterministic is:

$$Tree(G,r) \ := \ \{\ u \xrightarrow{a} ua \mid \exists\, s,\ r \underset{\underline{G}}{\overset{ua}{\Longrightarrow}} s \ \wedge\ a \in L\ \} \ \cup\ \{\ uf \mid r \underset{\underline{G}}{\overset{u}{\Longrightarrow}} s \ \wedge\ sf \in G\ \}$$

We also write $Tree(G) := Tree(G,r)$ when r is the unique root of G.

For instance, starting from the tree of Figure 3.1, we apply by inverse the finite mapping defined by $1 \ \mapsto\ \{f1a, f1g, g\overline{1}f\overline{2}f1a\}$, $2 \ \mapsto\ \{f2f, f\overline{2}f1a\}$ and $x \ \mapsto\ x$ for any colour $x \in \{f,g,a\}$, to get the coloured graph of the rightmost representation of figure below. Note that the rightmost graph of Figure 3.1 is a quotient of the tree: its root unfolding is the tree. To represent terms and their quotients, we consider a restriction of coloured graphs.

A *term graph* is a deterministic coloured graph labelled in $L = \mathbb{N} - \{0\}$ such that

$$\{\ i \mid \exists\ t,\ s \xrightarrow{i} t\ \} \quad = \quad \{1,\ldots,\varrho(c(s))\} \quad \text{for every vertex } s.$$

A general term graph is the graph $\overrightarrow{T(F)}$ of finite terms, defined by

$$\overrightarrow{T(F)} := \{\ ft_1\ldots t_{\varrho(f)} \xrightarrow{i} t_i \mid f \in F \ \wedge\ t_1,\ldots,t_{\varrho(f)} \in T(F) \ \wedge\ i \in [\varrho(f)]\ \}$$

$$\bigcup\ \{\ (ft_1\ldots t_{\varrho(f)})f \mid t_1,\ldots,t_{\varrho(f)} \in T(F) \ \wedge\ i \in [\varrho(f)]\ \}$$

and the *maximal quotient* $Graph(t)$ of any finite term t is

$$Graph(t) \ := \ \overrightarrow{T(F)}_{|\,\{\, s \,|\, t \,\Longrightarrow\, s\,\}}$$

the restriction of $\overrightarrow{T(F)}$ to its vertices accessible from t (the subterms of t).

For instance $Graph(fafgaa) \ = \ \{\, fafgaa \xrightarrow{1} a\,,\ fafgaa \xrightarrow{2} fgaa\,,$ $fgaa \xrightarrow{1} ga\,, fgaa \xrightarrow{2} a\,,\ ga \xrightarrow{1} a\,\}$ with the colouring $c(fafgaa) = c(fgaa) = f$, $c(ga) = g$, $c(a) = a$, and is represented by the rightmost figure in Figure 3.1.

A *term tree* is a term graph which is a tree, and we denote $TermTrees$ the family of term trees. Any finite term t is identified with the isomorphic class of the rooted unfolding of the maximal quotient of t: $Unf(Graph(t),t) \ = \ Unf(\overrightarrow{T(F)},t)$.

More generally, a *term* (finite or infinite) is the isomorphic class $[G]$ of a term tree G whose a standard canonical representative is $Tree(G)$. In particular, the canonical representative of any finite term t is

$$Tree(t) \ := \ Tree(Graph(t)) \ = \ Tree(\overrightarrow{T(F)},t)$$

For instance $Tree(fafgaa) \ = \ \{\,\varepsilon \xrightarrow{1} 1\,,\ \varepsilon \xrightarrow{2} 2\,, 2 \xrightarrow{1} 21\,, 21 \xrightarrow{1} 211\,,$ $2 \xrightarrow{2} 22\,\}$ with the colouring $c(\varepsilon) = c(2) = f$, $c(21) = g$, $c(1) = c(211) = c(22) = a$.

A well-known fact is that the family of the canonical (representatives of) term trees

$$Terms \ := \ \{ \ Tree(G) \mid G \in TermTrees \ \}$$

is a complete partial order by taking $\Omega \in F_0$ and by using the following partial order:

$$G \ \leq_\Omega \ H \quad \text{if} \quad G - V_G\{\Omega\} \ \subseteq \ H \quad \text{for any} \ \ G, H \in Terms$$

whose the smallest element is $\{\varepsilon\,\Omega\}$ and such that any increasing chain $G_0 \leq_\Omega \ldots \leq_\Omega G_n \leq_\Omega \ldots$ has a least upper bound

$$sup_{n\geq 0}\,(G_n) \ = \ (\bigcup_{n\geq 0} \underline{G_n}) \ \cup \ \{ \ uf \mid \exists\, m\ \forall\, n \geq m, \ uf \in G_n \ \}$$

For instance and for any $G \in Terms$, we define its *truncation* G_n to the level $n \geq 0$ by

$$G_n \ := \ \underline{G}_{|\,\{\,u\,|\,|u|\leq n\,\}} \cup \{ \ uf \in G \mid |u| < n \ \} \cup \{ \ u\,\Omega \mid \exists\, f, \ uf \in G \ \wedge \ |u| = n \ \}$$

to obtain an increasing chain $G_0 \ \leq_\Omega \ \ldots \ \leq_\Omega \ G_n \leq_\Omega \ \ldots$ of least upper bound G. So any mapping $h \ : \ T(F) \longrightarrow T(F)$ which is monotone

$$Tree(s) \ \leq_\Omega \ Tree(t) \quad \Longrightarrow \quad Tree(h(s)) \ \leq_\Omega \ Tree(h(t))$$

is extended into a continuous function $h: \ TermTrees \longrightarrow Terms$ by defining for any $G \in TermTrees$ and taking any $\Omega \ \in \ F_0 - Im(c_G)$

$$h(G) \ := \ sup_{n\geq 0}\, Tree(h(t_n)) \quad \text{if no node is labelled by } \Omega$$

where t_n is the unique finite term with $Tree(t_n) = (Tree(G))_n$; otherwise $h(G)$ is undefined.

As we want to produce only terms by inverse rational mappings with unfoldings from terms, we can restrict to any rational mapping h preserving the determinism:

$$G \ \text{deterministic} \quad \Longrightarrow \quad h^{-1}(G) \ \text{deterministic}$$

This implication remains true if for each $a \in L$, we restrict $h(a)$ to its minimal prefix subset of $(F(L \cup \overline{L}))^* F$: for any finite automaton recognizing $h(a)$, we determinize it, then we do the synchronized product with the automaton $\{0\}{\times}F{\times}\{1\} \ \cup \ \{1\}{\times}(L \cup \overline{L}){\times}\{0\}$ of initial state 0 and of final state 1, and then we remove any arc which is source of a final state. So we may assume that for each $a \in L$, $h(a)$ is recognized by a finite deterministic automaton (A, ι, T) such that each final state (in T) is terminal (source of no arc), and its initial state $\iota \in P := \{\, s \in V_A \mid s \xrightarrow[A]{F} \,\}$ which is disjoint of $\{\, s \in V_A \mid s \xrightarrow[A]{L\cup\overline{L}} \,\}$ and such that

$$s \in P \quad \Longleftrightarrow \quad t \in V_A - P \quad \text{for any transition} \ \ s \xrightarrow[A]{} t$$

A sufficient condition to have $h^{-1}(G)$ deterministic from a term tree G, is to add the following *determinism condition*:

$$s \xrightarrow[A]{a} \ \wedge \ s \xrightarrow[A]{b} \ \wedge \ a \in L \qquad \Longrightarrow \qquad a = b$$

and in that case, we say that h is a *deterministic rational mapping.*

For any graph family $\mathcal{F}$, we denote

$$\begin{aligned} DRat^{-1}(\mathcal{F}) \ := \ \{ \ h^{-1}(G) \mid G \in \mathcal{F} \ \wedge \ h : L \cup F {\longrightarrow} Rat((L \cup \overline{L} \cup F)^*) \\ \wedge \ h(F) \subseteq F \ \wedge \ |Dom(h)| < \infty \ \wedge \ h \text{ deterministic } \} \end{aligned}$$

the inverse deterministic rational mappings of graphs in $\mathcal{F}$. So we restrict the hierarchy $(Tree_n)$ on terms and by using only deterministic rational mappings:

$$Term_0 := \text{the set of regular term trees}$$

$$Term_{n+1} := Unf(DRat^{-1}(Term_n)) \ \cap \ TermTrees \ n \geq 0$$

where by commodity with the two next hierarchies, we start to $Tree_1$ restricted to terms. Recall that the terms over F_1 are the infinite words. Particular infinite words having a decidable monadic theory are the infinite *morphic words* [CT00] which are words of the form:

$$\sigma(\tau^\omega(a)) \quad = \quad \sigma(\, a\, u\, \tau(u) \ldots \tau^n(u) \ldots)$$

where σ and τ are morphisms from A^* into itself for some finite $A \subset F_1$ with $a \in A$ and $\tau(a) = au$. The decidability of the monadic theory for these infinite words is also a consequence that they are terms at level 2 of our hierarchy.

Proposition 3.2 *Any morphic word is in* $Term_2$ *.*

We consider the hierarchy of term families defined in [Da82] whose terms have a decidable monadic theory [CK02]. This hierarchy is defined as follows:

$Sub_0 :=$ the set of regular term trees

$Sub_{n+1} := \bigcup\{\ [Subst_{e,u}(Sub_n)] \mid u \in F_0^+ \wedge u(1) \neq \ldots \neq u(|u|) \wedge e \in F_{|u|+1}\,\}$

where for any finite term t, $Subst_{e,u}(t)$ is the finite term without e and is obtained by evaluating the function e as a *first order substitution*: its first argument is the term on which we apply the substitution and its $i+1$-th argument is the term which is substituted to $u(i)$ for each $1 \leq i \leq |u|$. Precisely $Subst_{e,u}(t)$ is defined by induction on the length of any finite term t as follows:

$Subst_{e,u}(ft_1 \ldots t_{\varrho(f)}) := f Subst_{e,u}(t_1) \ldots Subst_{e,u}(t_{\varrho(f)}) \quad$ if $\ f \neq e$

$Subst_{e,u}(et_0t_1 \ldots t_{|u|}) := Subst_{e,u}(t_0)[Subst_{e,u}(t_1)/u(1), \ldots, Subst_{e,u}(t_{|u|})/u(|u|)]$

where for any $n \geq 0, t, s_1, \ldots, s_n \in T(F), a_1 \neq \ldots \neq a_n \in F_0, t[s_1/a_1, \ldots, s_n/a_n]$ is the term obtained by simultaneous replacement in t of a_i by s_i (for $1 \leq i \leq n$), and is defined by induction on the length of t as follows:

$ft_1 \ldots t_{\varrho(f)}[s_1/a_1, \ldots, s_n/a_n] := f(t_1[s_1/a_1, \ldots, s_n/a_n]) \ldots (t_{\varrho(f)}[s_1/a_1, \ldots, s_n/a_n])$
if $f \notin \{a_1, \ldots, a_n\}$

$a_i[s_1/a_1, \ldots, s_n/a_n] := s_i \quad$ for every $\ i \in [n]$

Note that the mapping $Subst_{e,u} :\ T(F) \longrightarrow T(F - \{e\})$ is monotone for $\leq_\Omega$ if $\Omega \notin \{u(1), \ldots, u(|u|)\}$, and we extend by continuity $Subst_{e,u}$ to any term tree $G \in TermTrees$. Finally we extend by union $Subst_{e,u}$ to any set of term trees.

Theorem 3.3 *Starting from the regular terms, the terms generated by n first order substitutions are exactly the terms obtained by applying n inverse deterministic rational mappings each being followed by an unfolding:*

$$Sub_n \ = \ Term_n \quad \textit{for every} \ \ n \geq 0\,.$$

We consider now the extension of the first order substitution to the *second order substitution* $Subst_{e,r_1 \ldots r_n}$ where $e \in F_{n+1}$ and $r_1, \ldots, r_n$ are elementary terms *i.e.* of the form $f\, a_1 \ldots a_{\varrho(f)}$ with $f \in F$ and $a_1 \neq \ldots \neq a_{\varrho(f)} \in F_0$ and such that $r_1(1) \neq \ldots \neq r_n(1)$. This function $Subst_{e,r_1 \ldots r_n}$ is first defined by induction on the length of any finite term as follows:

$Subst_{e,r_1 \ldots r_n}(ft_1 \ldots t_{\varrho(f)}) := f Subst_{e,r_1 \ldots r_n}(t_1) \ldots Subst_{e,r_1 \ldots r_n}(t_{\varrho(f)})$ if $\ f \neq e$

and $\qquad Subst_{e,r_1 \ldots r_n}(et_0t_1 \ldots t_n)$

$\qquad := Subst_{e,r_1 \ldots r_n}(t_0)[Subst_{e,r_1 \ldots r_n}(t_1)/r_1, \ldots, Subst_{e,r_1 \ldots r_n}(t_n)/r_n]$

where for any $n \geq 0$, $t, s_1, \ldots, s_n \in T(F)$ and $r_1, \ldots, r_n \in T(F)$ elementary terms with $r_1(1) \neq \ldots \neq r_n(1)$, the term $t[s_1/r_1,\ldots,s_n/r_n]$ is defined by

$$\begin{aligned} & f t_1 \ldots t_{\varrho(f)}[s_1/r_1,\ldots,s_n/r_n] \\ := {} & f\,(t_1[s_1/r_1,\ldots,s_n/r_n])\ldots(t_{\varrho(f)}[s_1/r_1,\ldots,s_n/r_n]) \ \text{ if } \ f \notin \{r_1(1),\ldots,r_n(1)\} \\ := {} & s_i\,[\,(t_1[s_1/r_1,\ldots,s_n/r_n])/a_1\,,\ \ldots,\ (t_{\varrho(f)}[s_1/r_1,\ldots,s_n/r_n])/a_{\varrho(f)}\,] \\ & \hfill \text{if } \ r_i = f\,a_1\ldots a_{\varrho(f)} \end{aligned}$$

Note that $Subst_{e,r_1\ldots r_n} : T(F) \longrightarrow T(F-\{e\})$ is monotone for $\leq_\Omega$ if $\Omega \notin Occur(r_1) \cup \ldots \cup Occur(r_n)$, and similarly to the first order substitution, we extend by continuity the second order substitution $Subst_{e,r_1\ldots r_n}$ to any term tree. In particular $Subst_{e,r_1\ldots r_n}$ with $r_1,\ldots,r_n \in F_0$ corresponds to the first order substitution. Any second order substitution preserves the decidability of the monadic theory, and all the terms in $\bigcup_n Term_n$ are also the terms obtained from the regular terms by iterated applications of second order substitutions.

Proposition 3.4 *For any second order substitution $Subst_{e,r_1\ldots r_n}$ and any $q \geq 0$, $G \in Term_q \implies Subst_{e,r_1\ldots r_n}(G) \in Term_{q+2}$ when it is defined.*

We consider the hierarchy of families of terms which are least solutions of higher order recursive schemes [In76] [Da82] and that have a decidable monadic theory when the schemes are safe [KNU02]. By lack of space, it is not possible to reintroduce here the notion of a higher order recursive *scheme*, which is a deterministic grammar between typed terms. The level of a scheme is the maximum level of the left hand side types. The terms considered in [KNU02] are generated by schemes satisfying a safety condition. A scheme P is *safe* if for any rule $fa_1\ldots a_n \longrightarrow t$ there is no subterm s of t with an occurrence a_i in s of type level strictly less than the type level of s. The schemes of level 1 are safe, and have been first considered in [Ni75] and called *recursive program schemes* (see among others [Gu81] and [Co90]).

Theorem 3.5 *For every $n \geq 0$, the terms generated by the safe schemes of level at most n are exactly the terms in $Term_n$ i.e. the terms obtained from the regular terms by applying n inverse deterministic rational mappings each being followed by an unfolding.*

4 Conclusion

Several natural questions arise; we list some of them.

a) Let us begin with a question in [KNU02]. Does any solution of an unsafe scheme can be generated by a safe scheme, and in the negative case, has it a decidable monadic theory ?

b) Do we get more terms by allowing non deterministic rational mappings, and applying them not only on terms but on trees *i.e.*

$$\text{for every } n \geq 0,\ \ Term_n \ = \ Tree_n \ \cap \ TermTrees\ ?$$

c) Another question is to characterize the family of branch languages of all terms in

$$T := \textstyle\bigcup_n Term_n\,.$$

Recall that the branch languages for the terms in $Term_1$ is the family of deterministic context-free branch languages [Co83].

d) A last question follows from the fact that the hierarchy $(Term_n)$ coincides with the hierarchy of families of terms recognized by the deterministic pushdown automata with multi-level stacks. This last question is the equivalence problem for terms in T:

is the equality of terms in T decidable?

This problem is decidable for terms in $Term_1$ because it is inter-reducible to the equivalence problem of (one level) deterministic pushdown automata, which is decidable [Sen97].

References

[Ba98] K. Barthelmann, *When can an equational simple graph be generated by hyperedge replacement*, 23^{rd} MFCS, LNCS 1450, L. Brim, J. Gruska, J. Zlatuska (Eds.), 543–552 (1998).

[Bl02] A. Blumensath, *Axiomatising tree-interpretable structures*, 19^{th} STACS, LNCS 2285, H. Alt, A. Ferreira (Eds.), 596–607 (2002).

[CT00] O. Carton and W. Thomas, *The monadic theory of morphic infinite words and generalizations*, 25^{th} MFCS, LNCS 1893, M. Nielsen and B. Rovan (Eds.), 275–284 (2000).

[Ca90] D. Caucal, *On the regular structure of prefix rewriting*, 15^{th} CAAP, LNCS 431, A. Arnold (Ed.), 87–102 (1990) [a full version is in Theoretical Computer Science 106, 61–86 (1992)].

[Ca96] D. Caucal, *On infinite transition graphs having a decidable monadic theory*, 23^{rd} ICALP, LNCS 1099, F. Meyer auf der Heide, B. Monien (Eds.), 194–205 (1996) [a full version will appear in Theoretical Computer Science].

[CK01] D. Caucal and T. Knapik, *An internal presentation of regular graphs by prefix-recognizable ones*, Theory of Computing Systems 34-4 (2001).

[Co83] B. Courcelle, *Fundamental properties of infinite trees*, Theoretical Computer Science 25, 95–169 (1983).

[Co90] B. Courcelle, *Graph rewriting: an algebraic and logic approach*, Handbook of Theoretical Computer Science Vol. B, J. Leeuwen (Ed.), Elsevier, 193–242 (1990).

[Co90] B. Courcelle, *Recursive applicative program schemes*, Handbook of Theoretical Computer Science Vol. B, J. Leeuwen (Ed.), Elsevier, 459–492 (1990).

[Co94] B. Courcelle, *Monadic second-order definable graph transductions: a survey*, Theoretical Computer Science 126, 53–75 (1994).

[CK02] B. Courcelle and T. Knapik, *The evaluation of first-order substitution is monadic second-order compatible*, to appear in Theoretical Computer Science (2002).

[CW98] B. Courcelle and I. Walukiewicz, *Monadic second-order logic, graph coverings and unfoldings of transition systems*, Annals of Pure and Applied Logic 92, 35–62 (1998).

[Da82] W. Damm, *The IO and OI hierarchies*, Theoretical Computer Science 20 (2), 95–208 (1982).

[En83] J. Engelfriet, *Iterated push-down automata and complexity classes*, 15^{th} STOC, 365–373 (1983).

[Gu81] I. Guessarian, *Algebraic semantics*, LNCS 99, Sringer Verlag, (1981).

[In76] K. Indermark, *Schemes with recursion on higher types*, 5^{th} MFCS, LNCS 45, A. Mazurkiewicz (Ed.), 352–358 (1976).

[KNU01] T. Knapik, D. Niwiński and P. Urzyczyn, *Deciding monadic theories of hyperalgebraic trees*, 5^{th} International Conference on Typed Lambda Calculi and Applications, LNCS 2044, Abramsky (Ed.), 253–267 (2001).

[KNU02] T. Knapik, D. Niwiński and P. Urzyczyn, *Higher-order pushdown trees are easy*, 5^{th} FOSSACS, to appear in LNCS, M. Nielsen (Ed.), (2002).

[KV00] O. Kupferman and M. Vardi, *An automata-theoretic approach to reasoning about infinite-state systems*, 12^{th} CAV, LNCS 1855, A. Emerson, P. Sistla (Eds.), 36–52 (2000).

[LN01] S. La Torre and M. Napoli, *Automata-based representations for infinite graphs*, Theoretical Informatics and Applications 35, 311–330 (2001).

[MuS85] D. Muller and P. Schupp, *The theory of ends, pushdown automata, and second-order logic*, Theoretical Computer Science 37, 51–75 (1985).

[Ni75] M. Nivat, *On the interpretation of polyadic recursive schemes*, Symposia Mathematica 15, Academic Press (1975).

[Ra69] M. Rabin, *Decidability of second-order theories and automata on infinite trees*, Transactions of the American Mathematical Society 141, 1–35 (1969).

[Sem84] A. Semenov, *Decidability of monadic theories*, MFCS 84, LNCS 176, W. Brauer (Ed.), 162–175 (1984).

[Sen97] G. Sénizergues, *The equivalence problem for deterministic pushdown automata is decidable*, 24^{th} ICALP, LNCS 1256, P. Degano, R. Gorrieri, A. Marchetti-Spaccamela (Eds.), 671–681 (1997).

[Th90] W. Thomas, *Automata on infinite objects*, Handbook of Theoretical Computer Science Vol. B, J. Leeuwen (Ed.), Elsevier, 135–191 (1990).

[Th97] W. Thomas, *Languages, automata, and logic*, Handbook of Formal Languages, Vol. 3, G. Rozenberg, A. Salomaa (Eds.), Springer, 389–456 (1997).

[Ti86] J. Tiuryn, *Higher-order arrays and stacks in programming: An application of complexity theory to logics of programs*, 12^{th} MFCS, LNCS 233, J. Gruska, B. Rovan, J. Wiedermann (Eds.), 177–198 (1986).