PU. M. A. Vol. **22** (2011), No.1, pp. 1 – 38

Pattern language recognition and generation

Symeon Bozapalidis
Aristotle University of Thessaloniki
Department of Mathematics
54124, Thessaloniki, Greece
e-mail: bozapali@math.auth.gr

and
Antonios Kalampakas
Aristotle University of Thessaloniki
Department of Mathematics
54124, Thessaloniki, Greece
e-mail: akalamp@math.auth.gr

(Received: December 13, 2010)

Abstract. A k-bounded recognizable pattern language is the set of all k-bounded patterns that are obtained as behaviors of a pattern automaton. The emptiness and the equality problems for k-bounded behaviors of pattern automata are proved to be decidable. Moreover, we examine their relationship with context free pattern grammars, which are basically the hyperedge replacement grammars on planar directed acyclic graphs. An important result is that the classes of k-bounded recognizable and context free pattern languages are cones with respect to (k-bounded) pattern transductions. Pattern automata are also used to study 0-type grammars and word rewriting systems. It is proved that the emptiness problem is decidable for the set of k-bounded derivations of a 0-type grammar and that k-bounded confluence and k-bounded termination are decidable for word rewriting systems.

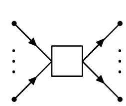
Mathematics Subject Classification(2010). 68R01, 68Q70, 05C85.

Keywords: automata, decidability, graph grammars, word rewriting systems.

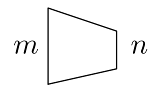
1 Introduction

Traditionally, a ranked symbol can be depicted either in a root to frontier way or in the dual way, frontier to root. The objects obtained by suitably combining the first (resp. second) such symbols are the well known root to frontier (resp. frontier to root) trees. The top down and bottom up tree automata are the machines that can recognize these trees respectively and they have been exhaustively investigated in the last years (cf. [15, 16]). On a higher complexity level are located the pdags (planar, directed, acyclic graphs) which are obtained from the elementary flowcharts

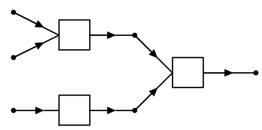
(Fl)



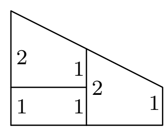
by means of the composition and boxing operations (cf. [17]). Automata recognizing such objects are studied in [5, 22, 23]. However, we have very poor results in this direction due to the complicated character of the considered machine models. Here we adopt a "lego" presentation for (Fl):



whose input and output faces have length m and n respectively. With this convention the pdags can be represented by figures that we call patterns. The composition and boxing operations are then interpreted as horizontal and vertical figure concatenation. For instance, the pdag



is simply represented by the following pattern.

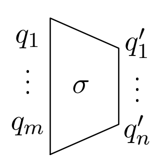


Fortunately, the set of all patterns has a nice algebraic structure, that of a magmoid, already used by Arnold and Dauchet in order to display a convenient setup for the study of trees (cf. [1, 2]). Roughly speaking, a magmoid is a doubly ranked family of sets $M = (M_{m,n})$ equipped with two operations (simulating the horizontal and vertical pattern concatenation):

$$\circ : M_{m,n} \times M_{n,k} \to M_{m,k}, \qquad m, n, k \geqslant 0,$$

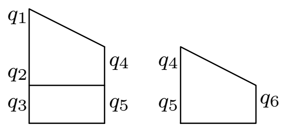
$$\square : M_{m,n} \times M_{m',n'} \to M_{m+m',n+n'}, \qquad m, n, m', n' \geqslant 0,$$

which are associative, unitary and compatible to each other. The set of patterns constructed from a doubly ranked alphabet Σ, denoted $mag(\Sigma)$, is actually the free magmoid over Σ and is playing the role that the free monoid plays for words. Our intention in this paper is to investigate pattern languages i.e., subsets of $mag(\Sigma)$. Two interesting classes of pattern languages are the behaviors of pattern automata and those generated by context free pattern grammars. The notion of a pattern automaton naturally generalizes that of a tree automaton (both top-down and bottom-up); such an automaton $\mathcal{A}$ consists of an input alphabet Σ, a state set Q and its moves are schemes of the form

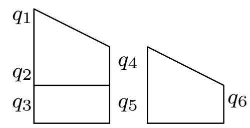
(mv)  $\sigma \in \Sigma_{m,n} \quad (m, n \geq 0).$

The behavior of $\mathcal{A}$ is obtained by forming patterns over these elements and taking care of two things:

- the state words occurring in opposite faces must be equal,
- the state words occurring as inputs and outputs must belong in two a priori given rational sets $I_{\mathcal{A}}$ and $T_{\mathcal{A}}$ of Q^*.



or simply



Then $|\mathcal{A}|$ is the set of patterns obtained by removing the frames of the state words. Bossut, Dauchet and Warin showed that one can choose one of $I_{\mathcal{A}}$ or $T_{\mathcal{A}}$ (but not both) to be of the simple form q^*, $q \in Q$. Behaviors of pattern automata, called recognizable pattern languages are closed under union, intersection and horizontal concatenation cf. [5]. The complicated recognition mechanism of the above automata has confined further research activity in this topic.

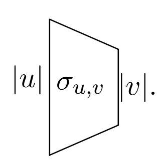
In this paper, we propose a bounded generation procedure for the above pattern automata. For a specified positive integer k, the k-bounded behavior of a pattern automaton $\mathcal{A}$ is obtained by starting and ending to state words of length $\leq k$, and also at each step keeping the rank of the constructed pattern $\leq k$. This mode of recognizing patterns is clearly non trivial and creates perspectives of achieving efficiency results. An important result of this paper states that both the emptiness and the equality problems, for k-bounded behaviors of pattern automata, are decidable.

In the context of patterns there are three types of star operations. Given a pattern language $L \subseteq mag(\Sigma)$ the language L° (resp. L) is obtained by horizontally (resp. vertically) concatenating the patterns of L and is called the $\circ$-star (resp. the $\square$-star) of L. The full star $L^{\bullet}$ of L, is the submagmoid of $mag(\Sigma)$ generated by L. Clearly $L^{\circ} \subset L^{\bullet}$ and L $\subset L^{\bullet}$. Actually $L^{\bullet}$ (and not L° or L) is the analog of the Kleene star of a word language. The class $Rec^{(k)}$ of k-bounded behaviors of pattern automata, or else k-bounded recognizable pattern languages, is shown to be closed under union, intersection, and also under the k-bounded full star operation.

The class of k-bounded recognizable pattern languages admitting a finite number of derivatives, called finite state languages, contains the class $Rec^{(k)}$. The equality problem is proved to be solvable for this class and moreover it is shown to be closed under boolean operations.

More than 30 years before, Benson, Schnorr, Claus and Hotz (cf. [3],[24], [9] and [20],[21]) used the Category theory formalism to describe phenomena in 0-type grammars which are instances of pattern automata. Actually, the set of derivations of a 0-type grammar is a pattern language. Here we prove that the emptiness problem is decidable for the set of k-bounded derivations of any given 0-type grammar.

On the other hand, rewriting word systems can be viewed as pattern automata of a specific form. Indeed, if $\mathcal{R} = (X, R)$ is a finite rewriting system, X the carrier set and R the set of rules, then a doubly ranked alphabet $\Sigma(\mathcal{R})$ can be constructed by associating to each rule $u \to v \in R$ the symbol



The pattern automaton $\mathcal{A}(\mathcal{R})$ with input alphabet $\Sigma(\mathcal{R})$, state set X, moves $(u, \sigma_{u,v}, v)$ and initial and final state sets $I_{\mathcal{A}(\mathcal{R})} = T_{\mathcal{A}(\mathcal{R})} = X^*$ has as behavior the set of all derivation patterns of the rewriting system $\mathcal{R}$.

Denoting by $\mathcal{R}(u,v)$ the set of all patterns starting from u and ending to v, the confluence condition in $\mathcal{R}$ can be formulated as follows:

$$\mathcal{R}(u,v_1) \neq \emptyset,\ \mathcal{R}(u,v_2) \neq \emptyset \quad \text{implies} \quad \mathcal{R}(v_1,v) \neq \emptyset,\ \mathcal{R}(v_2,v) \neq \emptyset$$

for some $v \in X^*$. Of course this condition is undecidable because the emptiness problem is undecidable for recursively enumerable languages which are projections of recognizable pattern languages under certain functions. Now if, in this setup, we adopt k-bounded rewriting processes, the above confluence problem is decidable because the language $\mathcal{R}^{(k)}(u,v)$ of all k-bounded derivation patterns from u to v is k-bounded recognizable.

By an analogous way we show that k-bounded termination is decidable for a finite rewriting system $\mathcal{R}$, also we can decide whether a finite rewriting system is k-bounded equivalent (i.e., computes the same k-bounded relation) with any of its proper subsystems.

Furthermore, in this paper, we investigate context free pattern grammars, which are identical with the well known hyperedge replacement grammars (cf. [11, 12, 18, 19]) by viewing a pattern as a planar directed acyclic graph. It turns out that every context free pattern language is k-bounded and also that the intersection of a context free with a recognizable (or k-bounded recognizable) is again context free. Moreover, this class contains the class of k-bounded recognizable languages and it is closed under pattern homomorphisms and inverse alphabetic homomorphisms.

Given doubly ranked alphabets Σ and Σ', a pattern transducer from Σ to Σ' is just a pattern automaton whose moves are triples $(u, \sigma/p_\sigma, v)$ with u, v state words, $\sigma \in \Sigma$ and p_σ a pattern in $mag(\Sigma')$. Its behavior (k-bounded behavior) is then a relation from $mag(\Sigma)$ to $mag(\Sigma')$ ($mag^{(k)}(\Sigma)$ to $mag^{(k)}(\Sigma')$) called a *Rec* ($Rec^{(k)}$) -transduction. Instances of transductions of this kind are the intersection with a recognizable (k-bounded) pattern language, arbitrary (k-bounded) homomorphisms and inverse alphabetic homomorphisms.

A *Rec*-cone ($Rec^{(k)}$-cone) is a class of pattern languages closed under *Rec*-transductions ($Rec^{(k)}$-transductions). A main result of this paper is that the classes of k-bounded recognizable and context free pattern languages are $Rec^{(k)}$-cones.

2 Preliminaries

2.1 Magmoids

Recall that a doubly ranked set (or a doubly ranked alphabet) $(A_{m,n})_{m,n\in\mathbb{N}}$ is a set A together with a function $rank : A \to \mathbb{N} \times \mathbb{N}$, where $\mathbb{N}$ is the set of natural numbers. For $m, n \in \mathbb{N}$, $A_{m,n}$ is the set $\{a \in A \mid rank(a) = (m,n)\}$. In what follows we will drop the subscript $m, n \in \mathbb{N}$ and denote a doubly ranked set simply by $(A_{m,n})$.

A *semi-magmoid* is a doubly ranked set $M = (M_{m,n})$ equipped with two operations

$$\circ : M_{m,n} \times M_{n,k} \to M_{m,k}, \qquad m,n,k \geqslant 0$$

$$\square : M_{m,n} \times M_{m',n'} \to M_{m+m',n+n'}, \qquad m,n,m',n' \geqslant 0$$

which are associative in the obvious way and satisfy the distributivity law

$$(f \circ g) \,\square\, (f' \circ g') = (f \,\square\, f') \circ (g \,\square\, g')$$

whenever all the above operations are defined.

A *magmoid* is a semi-magmoid $M = (M_{m,n})$, equipped with a sequence of constants $e_n \in M_{n,n}$ ($n \geqslant 0$), called units, such that

$$e_m \circ f = f = f \circ e_n, \quad e_0 \,\square\, f = f = f \,\square\, e_0$$

for all $f \in M_{m,n}$ and all $m, n \geqslant 0$, and the additional condition

$$e_m \,\square\, e_n = e_{m+n}, \qquad \text{for all } m, n \geqslant 0$$

holds.

Notice that, due to the last equation, the element e_n ($n \geq 2$) is uniquely determined by e.

Semi-magmoids, morphisms, congruences and quotients of magmoids are defined in the obvious way.

In order to fix our notation we need the following definitions. Let $M = M_{m,n}$ be a magmoid. We say that a doubly ranked family $L = (L_{m,n})$ is a *subset* of M (notation $L \subseteq M$), whenever $L_{m,n} \subseteq M_{m,n}$ for all $m, n \in \mathbb{N}$. The boolean operations on subsets of M are defined in the following way: for $L, L' \subseteq M$ we set

$$(L \cup L')_{m,n} = L_{m,n} \cup L'_{m,n}, \;\; (L \cap L')_{m,n} = L_{m,n} \cap L'_{m,n}, \;\; (\bar{L})_{m,n} = \overline{(L_{m,n})}$$

where the operator "$^-$" stands for the set theoretic complement.

Next, given subsets L, L' of a magmoid M (with unit sequence e_n) we define their $\circ$ *-product* $L \circ L'$ by setting

$$(L \circ L')_{m,n} = \bigcup_{k \geqslant 0} L_{m,k} \circ L'_{k,n}, \qquad m, n \in \mathbb{N}$$

and their $\square$ *-product* $L \,\square\, L'$ by setting

$$(L \,\square\, L')_{m,n} = \bigcup_{\substack{\kappa+\kappa'=m \\ \lambda+\lambda'=n}} L_{\kappa,\lambda} \,\square\, L'_{\kappa',\lambda'}, \qquad m, n \in \mathbb{N}.$$

The $\circ$ *-star* is the union of the successive $\circ$ *-powers* of $L \subseteq M$:

$$L^{\circ} = \bigcup_{k \geqslant 0} L^{\circ,k}$$

where $L^{\circ,k}$ is inductively given by

$$L^{\circ,0} = E, \quad L^{\circ,1} = L, ..., L^{\circ,k+1} = L \circ L^{\circ,k}.$$

The $\square$ *-star* L is defined analogously.

The most important star operator in pattern language theory associates to any pattern language $L \subseteq mag(\Sigma)$ the pattern language $L^{\bullet}$, called the full star of L, which is the submagmoid of $mag(\Sigma)$ generated by L. It is exactly the two dimensional analog for the well known Kleene star for word languages. Indeed, for any doubly ranked alphabet Σ, we have $\Sigma^{\bullet} = mag(\Sigma)$, while the inclusions $\Sigma^{\circ} \subset mag(\Sigma)$, Σ $\subset mag(\Sigma)$ are proper.

EXAMPLE 2.1 (MAGMOIDS OF FUNCTIONS AND RELATIONS) The sets $Funct_{m,n}(Q)$ of all functions from Q^m to Q^n

$$Funct_{m,n}(Q) = \{f \mid f : Q^m \to Q^n\}, \qquad m,n \geqslant 0,$$

can be structured into a magmoid with $\circ$ being the usual function composition, while the operation $\Box$ is the function boxing defined as follows: for $f \in Funct_{m,n}(Q)$ and $f' \in Funct_{m',n'}(Q)$

$$f \Box f'(u \Box u') = f(u) \Box f'(u'), \qquad u \in V_m(Q)\,,\, u' \in V_{m'}(Q).$$

In a similar way the sets

$$Rel_{m,n}(Q) = \{R \mid R \subseteq Q^m \times Q^n\}$$

of all relations from Q^m to Q^n can be organized into a magmoid, $\circ$ being the relation composition and $\Box$ the relation concatenation.

Clearly $Funct(Q) = (Funct_{m,n}(Q))$ is a sub-magmoid of $Rel(Q) = (Rel_{m,n}(Q))$.

EXAMPLE 2.2 (MAGMOIDS OF TREES) Let $\Gamma = \bigcup_{k \geq 0} \Gamma_k$ be a ranked alphabet and $X = \{x_1, x_2, \dots\}$ a set of variables. We put $X_k = \{x_1, \dots, x_k\}$, $X_0 = \emptyset$. The set of trees $T_\Gamma(X_k)$ is the smallest set inductively defined by the items

i) $\Gamma_0 \cup X_k \subseteq T_\Gamma(X_k)$,

ii) if $f \in \Gamma_n$ and $t_1, \dots, t_n \in T_\Gamma(X_k)$, then $f(t_1, \dots, t_n) \in T_\Gamma(X_k)$.

The basic operation on trees is substitution which to every $t \in T_\Gamma(X_m)$ and $t_1, \dots, t_m \in T_\Gamma(X_n)$ associates the tree

$$t[t_1, \dots, t_m] \in T_\Gamma(X_n)$$

obtained by replacing t_i at all occurrences of x_i in t $(1 \leq i \leq m)$. Clearly substitution is associative

$$t[t_1, \dots, t_m][t'_1, \dots, t'_k] = t[t_1[t'_1, \dots t'_k], \dots, t_m[t'_1, \dots, t'_k]]$$

(whenever defined) and the following two unitary identities are trivially fulfilled

$$t[x_1, \dots, x_n] = t, \ (t \in T_\Gamma(X_n)) \text{ and } x_i[t_1, \dots, t_k] = t_i, \ 1 \leq i \leq k.$$

Now the sets $T_\Gamma(X)_{m,n} = (T_\Gamma(X_n))^m$ $(m,n \geq 0)$ are organized into a magmoid by defining the $\circ$- and $\Box$-operation

$$\circ : T_\Gamma(X_n)^m \times T_\Gamma(X_k)^n \to T_\Gamma(X_k)^m,$$

$$\Box : T_\Gamma(X_{n_1})^{m_1} \times T_\Gamma(X_{n_2})^{m_2} \to T_\Gamma(X_{n_1+n_2})^{m_1+m_2},$$

by the formulas:

$$(Op) \qquad (t_1, \dots, t_m) \circ (t'_1, \dots, t'_n) = (t_1[t'_1, \dots, t'_n], \dots, t_m[t'_1, \dots, t'_n])$$

$$(t_1, \dots, t_{m_1}) \Box (t'_1, \dots, t_{m_2}) = (t_1 \dots, t_{m_1}, t'_1, \dots, t_{m_2}).$$

Notice that $T_\Gamma(X)$ is not free in the category of magmoids.

Let Σ be a doubly ranked alphabet. We denote by $smag(\Sigma) = (smag_{m,n}(\Sigma))$ the smallest doubly ranked set satisfying the next items:

- $\Sigma_{m,n} \subseteq smag_{m,n}(\Sigma)$ for all $m, n \geq 0$,
- if $p \in smag_{m,n}(\Sigma)$ and $q \in smag_{n,k}(\Sigma)$ then their horizontal concatenation $p\,q \in smag_{m,k}(\Sigma)$,
- if $p \in smag_{m,n}(\Sigma)$ and $p' \in smag_{m',n'}(\Sigma)$ then their vertical concatenation $\begin{pmatrix} p \\ p' \end{pmatrix} \in smag_{m+m',n+n'}(\Sigma)$.

It is easy to see that $smag(\Sigma)$ is a semi-magmoid. Denote by $\varepsilon, \varepsilon_0$ the horizontal and vertical empty words respectively; the element

$$\varepsilon_n = \begin{pmatrix} \varepsilon \\ \vdots \\ \varepsilon \end{pmatrix} \qquad n \text{ times}$$

(called empty word of rank n) has obviously the property

$$\varepsilon_m\, p = p = p\, \varepsilon_n$$

for all $p \in smag_{m,n}(\Sigma)$. By convention, the empty word of rank 0 is identified with the vertical empty word ε_0.

The set $mag(\Sigma) = smag(\Sigma \cup \{\varepsilon, \varepsilon_0\})$ is by construction a magmoid. Actually it is the free magmoid generated by Σ as confirms the next theorem.

Theorem 2.3 *For every magmoid $M = (M_{m,n})$ and every doubly ranked function $f : \Sigma \to M$, there exists a unique morphism of magmoids $\bar{f} : mag(\Sigma) \to M$ making the following triangle commutative.*

$$\begin{array}{ccc} \Sigma & \xrightarrow{\;j\;} & mag(\Sigma) \\ & \searrow^{f} & \downarrow{\bar{f}} \\ & & M \end{array} \qquad j(x) = x, \;\; x \in \Sigma$$

Actually, $\bar{f}$ is given by the clauses,

- $\bar{f}(\sigma) = f(\sigma)$, *for all* $\sigma \in \Sigma$,
- $\bar{f}(\varepsilon_n) = e_n$, $n \geq 0$, *where* (e_n) *is the unit sequence of* M,
- $\bar{f}(p\,q) = \bar{f}(p) \circ \bar{f}(q)$, $\bar{f}\begin{pmatrix} p \\ p' \end{pmatrix} = \bar{f}(p) \,\square\, \bar{f}(p')$,

for all $p, q, p' \in smag(\Sigma)$ of suitable rank.

The elements of $mag_{m,n}(\Sigma)$ are called (m, n)-patterns over Σ. Subsets of $mag(\Sigma)$ are called *pattern languages*. Our patterns are exactly the unsorted abstract dags of [22],[23] and [5]. For another formalization see also [17].

Given alphabets Σ, Σ' a *pattern homomorphism* from Σ to Σ' is just a function $h : \Sigma \to mag(\Sigma')$; by virtue of the previous theorem, h can be uniquely extended into a morphism of magmoids

$$\bar{h} : mag(\Sigma) \to mag(\Sigma')$$

defined by

- $\bar{h}(\sigma) = h(\sigma)$, $\sigma \in \Sigma$,
- $\bar{h}(\varepsilon_n) = \varepsilon'_n$, $n \geq 0$,
- $\bar{h}(p_1 p_2) = \bar{h}(p_1)\bar{h}(p_2)$, $\quad \bar{h}\begin{pmatrix} p_1 \\ p_2 \end{pmatrix} = \begin{pmatrix} \bar{h}(p_1) \\ \bar{h}(p_2) \end{pmatrix}$,

for all p_1, p_2 of suitable rank. We often denote $\bar{h}$ simply by h when no confusion is caused.

A pattern homomorphism $h : mag(\Sigma) \to mag(\Sigma')$ is said to be

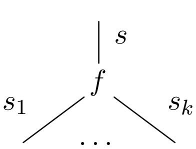
- *non erasing*, if for all $n \geq 0$ and all $\sigma \in \Sigma_{n,n}$, we have $h(\sigma) \neq \varepsilon_n$;
- *alphabetic*, whenever: $h(\Sigma_{n,n}) \subseteq \Sigma'_{n,n} \cup \varepsilon_n$, for all $n \geq 0$ and $h(\Sigma_{m,n}) \subseteq \Sigma'_{m,n}$, for all $m, n \geq 0$, $m \neq n$;
- *strictly alphabetic*, whenever $h(\Sigma_{m,n}) \subseteq \Sigma'_{m,n}$, for all $m, n \geq 0$.

2.2 Many Sorted Alphabets

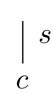
Let S be a set of sorts. An S-sorted alphabet is a set Γ together with a rank function

$$rank : \Gamma \to S^* \times S.$$

The set $\Gamma_{w,s} = \{f \mid f \in \Gamma, rank(f) = (w, s)\}$, $(w, s) \in S^* \times S$, is the set of symbols with profile $w \to s$. An element $f \in \Gamma_{w,s}$ is depicted as

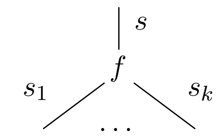


if $w = s_1 \cdots s_k$, or

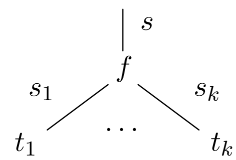


if $w = \varepsilon$. The set T^s_Γ of Γ-trees of sort s is the smallest set satisfying the following items:

- $\Gamma_{\varepsilon,s} \subseteq T^s_\Gamma$, $(s \in S)$,
- if



is an element of $\Gamma_{s_1\cdots s_k,s}$ and $t_i \in T_\Gamma^{s_i}$, $(1 \le i \le k)$, then



is in T_Γ^s.

Often in order to simplify the notation we write $f(t_1, \ldots, t_k)$ for the above tree.

The *height* of a tree $t \in T_\Gamma^s$ is defined by

- $height(c) = 0$, $c \in \Gamma_{\varepsilon,s}$, $(s \in S)$,
- $height(f(t_1, \ldots, t_k)) = 1 + max\{height(t_i) \mid 1 \le i \le k\}$.

A *morphism* from the S-sorted alphabet Γ to the S'-sorted alphabet Γ' is a pair $\Phi = (\varphi_0, \varphi)$ consisting of a sort function $\varphi_0 : S \to S'$ and a family of symbol functions

$$\varphi_{w,s} : \Gamma_{w,s} \to \Gamma'_{\varphi_0^*(w),\varphi_0(s)}, \quad (w \in S^*,\ s \in S),$$

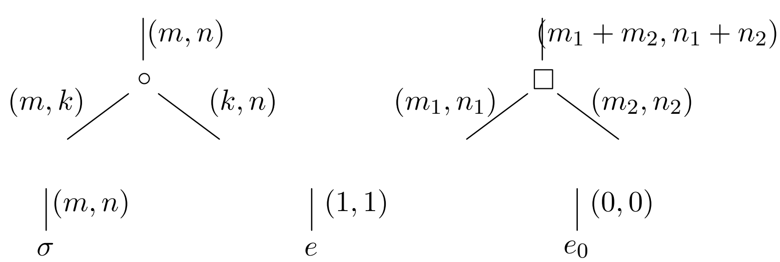
where $\varphi_0^* : S^* \to S'^*$ is the monoid morphism canonically extending φ_0. Such a pair induces a function

$$\Phi_s : T_\Gamma^s \to T_{\Gamma'}^{\varphi_0(s)}, \qquad s \in S,$$

according to the inductive formulas

- $\Phi_s(c) = \varphi_{\varepsilon,s}(c)$, $c \in \Gamma_{\varepsilon,s}$,
- $\Phi_s(f(t_1, \ldots, t_k)) = \varphi_{s_1\cdots s_k,s}(f)(\Phi_{s_1}(t_1), \ldots, \Phi_{s_k}(t_k))$, for $f \in \Gamma_{s_1\cdots s_k,s}$ and $t_i \in T_\Gamma^{s_i}$, $(1 \le i \le k)$.

For a given doubly ranked alphabet $\Sigma = (\Sigma_{m,n})$ we introduce the $\mathbb{N}\times\mathbb{N}$-sorted alphabet $\Gamma(\Sigma)$ consisting of all symbols of the following form.



There results a canonical function

$$h_\Sigma^{(m,n)} : T_{\Gamma(\Sigma)}^{(m,n)} \to mag_{m,n}(\Sigma), \qquad m, n \in \mathbb{N},$$

by setting

- $h_\Sigma^{(m,n)}(\sigma) = \sigma$, $(\sigma \in \Sigma_{m,n}, m, n \geq 0)$,
- $h_\Sigma^{(1,1)}(e) = \varepsilon$, $h_\Sigma^{(0,0)}(e_0) = \varepsilon_0$, where ε, ε_0 are the unit elements of $mag(\Sigma)$,
- $h_\Sigma^{(m,n)}(t_1 \circ t_2) = h_\Sigma^{(m,k)}(t_1)\, h_\Sigma^{(k,n)}(t_2)$,
- $h_\Sigma^{(m_1+m_2,n_1+n_2)}(t_1 \,\square\, t_2) = \begin{pmatrix} h_\Sigma^{(m_1,n_1)}(t_1) \\ h_\Sigma^{(m_2,n_2)}(t_2) \end{pmatrix}$.

We define the *height* of a pattern $p \in mag_{m,n}(\Sigma)$ to be the number $height(p) = min\{height(t) \mid h_\Sigma^{(m,n)}(t) = p\}$.

Given an integer $k \geq 1$, a tree $t \in T_{\Gamma(\Sigma)}^{(m,n)}$ is *k-bounded* if all the sorts occurring in t are $\leq k$ (hence $m, n \leq k$). The set of k-bounded trees is denoted $T_{\Gamma(\Sigma)}^{(k)}$. A pattern $p \in mag(\Sigma)$ is k-bounded if $p = h_\Sigma(t)$, for some $t \in T_{\Gamma(\Sigma)}^{(k)}$. The set of all k-bounded patterns over Σ is denoted $mag^{(k)}(\Sigma)$.

A language $L \subseteq mag(\Sigma)$ is called k-bounded whenever all its elements can be represented by k-bounded trees in $T_\Gamma(\Sigma)$. It is important to emphasize that for k-bounded languages $L, L' \subseteq mag(\Sigma)$ the language $L \circ L'$ is still k-bounded, while this is not generally the case for $L \,\square\, L'$. So, we speak of the k-bounded $\square$-product of L and L'. It is the k-bounded part of $L \,\square\, L'$:

$$L \,\square_k\, L' = (L \,\square\, L') \cap mag^{(k)}(\Sigma).$$

Similarly the k-bounded full star of a given language L is the k-bounded part of $L^\bullet$:

$$L^{\bullet_k} = L^\bullet \cap mag^{(k)}(\Sigma).$$

A pattern homomorphism $h : mag(\Sigma) \to mag(\Sigma')$ is said to be k-bounded whenever it sends the k-bounded symbols of Σ to k-bounded patterns of Σ, i.e., for all $\sigma \in \Sigma_{m,n}$ $(m, n \leq k)$, we have $h(\sigma) \in mag^{(k)}(\Sigma')$.

LEMMA 2.4 *Every k-bounded homomorphism $h : mag(\Sigma) \to mag(\Sigma')$ sends k-bounded patterns over Σ to k-bounded patterns over Σ', i.e., $h(mag^{(k)}(\Sigma)) \subseteq mag^{(k)}(\Sigma')$.*

Proof. Let $p \in mag^{(k)}(\Sigma)$ and $t \in T_{\Gamma(\Sigma)}^{(k)}$ so that $h_\Sigma(t) = p$; in order to get the image $h(p)$ we only have to replace every leaf $\sigma \in \Sigma_{m,n}$ of t, by the pattern $h(\sigma)$. Since, by assumption, $h(\sigma)$ is k-bounded, there exists $t_\sigma \in T_{\Gamma(\Sigma')}^{(k)}$ so that $h_{\Sigma'}(t_\sigma) = h(\sigma)$. The result follows. □

The inverse image of a k-bounded language via a k-bounded homomorphism needs not to be k-bounded, as it is illustrated in the following example.

EXAMPLE 2.5 Consider the alphabets $\Sigma_{3,3} = \{\sigma\}$, $\Sigma_{12} = \{\sigma_{12}\}$, $\Sigma_{21} = \{\sigma_{21}\}$ and $\Gamma_{1,1} = \{\gamma\}$, $\gamma_{12} = \{\gamma_{12}\}$, $\Gamma_{21} = \{\gamma_{21}\}$ and the 2-bounded homomorphism $h : mag(\Sigma) \to mag(\Gamma)$ defined by: $h(\sigma_{12}) = \gamma_{12}$, $h(\sigma_{21}) = \gamma_{21}$ and

$$h(\sigma) = \begin{pmatrix} \gamma \\ \gamma \\ \gamma \end{pmatrix}.$$

Then h sends the non 2-bounded pattern

$$\sigma_{12}\begin{pmatrix} e \\ \sigma_{12}\end{pmatrix}\begin{pmatrix} e_2 \\ \sigma_{12}\end{pmatrix}\begin{pmatrix} e_3 \\ \sigma_{12}\end{pmatrix}\begin{pmatrix} e_4 \\ \sigma_{12}\end{pmatrix}\begin{pmatrix} \sigma \\ \sigma\end{pmatrix}\begin{pmatrix} e_4 \\ \sigma_{21}\end{pmatrix}\begin{pmatrix} e_3 \\ \sigma_{21}\end{pmatrix}\begin{pmatrix} e_2 \\ \sigma_{21}\end{pmatrix}\begin{pmatrix} e \\ \sigma_{21}\end{pmatrix}\sigma_{21}$$

to the pattern

$$\gamma_{12}\begin{pmatrix} e \\ \gamma_{12}\end{pmatrix}\begin{pmatrix} e_2 \\ \gamma_{12}\end{pmatrix}\begin{pmatrix} e_3 \\ \gamma_{12}\end{pmatrix}\begin{pmatrix} e_4 \\ \gamma_{12}\end{pmatrix}\begin{pmatrix} \gamma \\ \vdots \\ \gamma\end{pmatrix}\begin{pmatrix} e_4 \\ \gamma_{21}\end{pmatrix}\begin{pmatrix} e_3 \\ \gamma_{21}\end{pmatrix}\begin{pmatrix} e_2 \\ \gamma_{21}\end{pmatrix}\begin{pmatrix} e \\ \gamma_{21}\end{pmatrix}\gamma_{21}$$

which is a 2-bounded pattern (we only have to apply the distributivity law in the definition of a magmoid, see subsection 2.1).

Given an S'-sorted alphabet Γ, we denote by $(P_\Gamma)_s^{s'}$ the set of all pruned trees of the form τ:

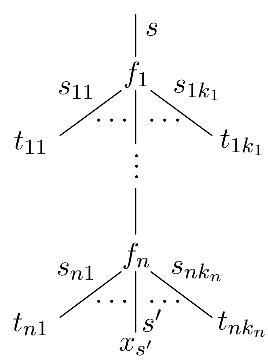


Figure 1

where $x_{s'}$ is a symbol of sort s'. There is a canonical function

$$(P_\Gamma)_{s'}^s \times T_\Gamma^{s'} \to T_\Gamma^s, \qquad (\tau, t) \mapsto \tau t$$

which sends every τ in $(P_\Gamma)_{s'}^s$ (as in Figure 1) and every $t \in T_\Gamma^{s'}$ to the tree of T_Γ^s obtained by replacing the variable $x_{s'}$ in τ by t.

3 Pattern Automata

Pattern automata have originated in [10], see also [24], and their basic properties are examined in [22],[23] and [5] under the name of pdag automata. A *pattern automaton* is a structure

$$\mathcal{A} = (\Sigma, Q, \theta_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$$

where Σ, Q are the finite sets of input alphabet and states respectively, $I_{\mathcal{A}}, T_{\mathcal{A}}$ are rational subsets of the monoid Q^* (i.e., $I_{\mathcal{A}}, T_{\mathcal{A}} \subseteq Rat(Q^*)$) and

$$\theta_{\mathcal{A}} : \Sigma \to Rel(Q)$$

is the move function. Its *behavior* is the pattern language

$$|\mathcal{A}| = \{p \mid p \in mag_{m,n}(\Sigma),\ \bar{\theta}_{\mathcal{A}}(p) \cap (I_{\mathcal{A}}^{(m)} \times T_{\mathcal{A}}^{(n)}) \neq \emptyset,\ m,n \in \mathbb{N}\}$$

where $\bar{\theta}_{\mathcal{A}} : mag(\Sigma) \to Rel(Q)$ is the unique magmoid morphism extending $\theta_{\mathcal{A}}$ and $I_{\mathcal{A}}^{(m)} = I_{\mathcal{A}} \cap Q^m$, $T_{\mathcal{A}}^{(n)} = T_{\mathcal{A}} \cap Q^n$.

Subsets of $mag(\Sigma)$ obtained as behaviors of pattern automata over Σ are called *recognizable* and their set is denoted $Rec(\Sigma)$. The k-bounded behavior of the pattern automaton $\mathcal{A}$ is the pattern language

$$|\mathcal{A}|^{(k)} = |\mathcal{A}| \cap mag^{(k)}(\Sigma).$$

Call a pattern language L, *k-bounded recognizable*, whenever $L = |\mathcal{A}|^{(k)}$ for some pattern automaton $\mathcal{A}$. The class of all k-bounded recognizable pattern languages is denoted $Rec^{(k)}(\Sigma)$.

If we replace in the definition of a pattern automaton the magmoid $Rel(Q)$ of relations over Q by the magmoid $Funct(Q)$ of functions over Q, then we obtain the notion of a deterministic pattern automaton.

As for ordinary automata, the behaviors of pattern automata can be obtained as projections of *local pattern languages*.

Let us introduce the doubly ranked alphabet $\mathbf{Q}$ with

$$\mathbf{Q}_{m,n} = Q^m \times Q^n, \quad m,n \geqslant 0,$$

as well as the product alphabet $\Sigma \times \mathbf{Q}$ with

$$(\Sigma \times \mathbf{Q})_{m,n} = \Sigma_{m,n} \times \mathbf{Q}_{m,n}, \quad m,n \geqslant 0.$$

We denote by $pr_\Sigma : mag(\Sigma \times \mathbf{Q}) \to mag(\Sigma)$ the unique morphism extending the canonical projection

$$(u, \sigma, v) \mapsto \sigma.$$

The set $Comp(\mathcal{A}) \subseteq mag(\Sigma \times \mathbf{Q})$ of computations of $\mathcal{A}$ and the *i-th face* function $\partial_i : Comp(\mathcal{A}) \to Q^*$ $(i = 0, 1)$ are simultaneously defined inductively as follows:

- $(u, \sigma, v) \in Comp(\mathcal{A})$, and $\partial_0(u, \sigma, v) = u$, $\partial_1(u, \sigma, v) = v$, for all $(u, v) \in \theta_{\mathcal{A}}(\sigma)$, $\sigma \in \Sigma$, and $(q, \varepsilon, q) \in Comp(\mathcal{A})$, for all $q \in Q$, and $\partial_i(q, \varepsilon, q) = q$, $i = 1, 2$;

- if $p \in Comp_{m,n}(\mathcal{A})$, $q \in Comp_{n,k}(\mathcal{A})$ and $\partial_1(p) = \partial_0(q)$ then,

$$pq \in Comp_{m,k}(\mathcal{A}) \text{ and } \partial_0(pq) = \partial_0(p),\ \partial_1(pq) = \partial_1(q);$$

- if $p \in Comp_{m,n}(\mathcal{A})$, $p' \in Comp_{m',n'}(\mathcal{A})$ then,

$$\begin{pmatrix} p \\ p' \end{pmatrix} \in Comp_{m+m',n+n'}(\mathcal{A}) \text{ and}$$

$$\partial_0 \begin{pmatrix} p \\ p' \end{pmatrix} = \partial_0(p)\partial_0(p'), \quad \partial_1 \begin{pmatrix} p \\ p' \end{pmatrix} = \partial_1(p)\partial_1(p').$$

The local set $Loc(\mathcal{A})$ is the set of all computations $p \in Comp(\mathcal{A})$ such that $\partial_0(p) \in I_{\mathcal{A}}$ and $\partial_1(p) \in T_{\mathcal{A}}$. By construction,

$$|\mathcal{A}| = pr_\Sigma(Loc(\mathcal{A})).$$

A *successful run* of a pattern $p \in mag(\Sigma)$ in $\mathcal{A}$ is an element of the set $run_{\mathcal{A}}(p)$, where $run_{\mathcal{A}}$ denotes the next composition

$$(c) \qquad mag(\Sigma) \xrightarrow{pr_\Sigma^{-1}} mag(\Sigma \times \mathbf{Q}) \xrightarrow{\_ \cap Loc(\mathcal{A})} mag(\Sigma \times \mathbf{Q}) \xrightarrow{pr_{\mathbf{Q}}} mag(\mathbf{Q}).$$

where $pr_{\mathbf{Q}}$ is the projection $(u, \sigma, v) \mapsto (u, v)$.

It follows that

$$|\mathcal{A}| = \{p \mid p \in mag(\Sigma),\ run(p) \neq \emptyset\}.$$

It should be noticed, that the construction of the sets of computations and runs of an automaton $\mathcal{A}$ can be made for the k-bounded case: we only have to respect the k-bound in the formation of the patterns of $Comp(\mathcal{A})$ and $Loc(\mathcal{A})$. The corresponding sets are denoted $Comp^{(k)}(\mathcal{A})$ and $Loc^{(k)}(\mathcal{A})$.

Models of ordinary automata can be fit in the above framework:

Word Automata. Let A be an ordinary alphabet and denote by $\Sigma(A)$ the doubly ranked alphabet with $\Sigma_{1,1}(A) = A$, $\Sigma_{m,n}(A) = \emptyset$, for $m \neq 1$ or $n \neq 1$. Let $\mathcal{M} = (Q, \delta_{\mathcal{M}}, I, T)$ be an ordinary A-automaton where $\delta_{\mathcal{M}} \subseteq Q \times A \times Q$ is the move function and $I, T \subseteq Q$ are the initial and final state sets respectively. Its behavior is a word language $|\mathcal{M}| \subseteq A^*$. Clearly $\mathcal{M}$ can be viewed as a pattern automaton over $\Sigma(A)$. By allowing I and T to be rational subsets of Q^*, the behavior of $\mathcal{M}$ is a pattern language of $mag(\Sigma(A))$. For instance, if we have the automaton

$$\mathcal{M} : q_0 \underset{b}{\overset{a}{\rightleftarrows}} q_1, \qquad I = \{q_0\},\ T = \{q_1\},$$

its behavior is the word language $L = a(ba)^* \subseteq \{a, b\}^*$. By choosing $I = \{q_0^2\}$ and $T = \{q_1^2\}$ the corresponding behavior of $\mathcal{M}$ is the pattern language

$$\begin{pmatrix} L \\ L \end{pmatrix} = \{ \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \mid w_1, w_2 \in L\}.$$

By choosing $I = q_0^* = T$ the corresponding behavior is the $\square$-star of the language $(ab)^*$.

Tree Automata. Given a finite ranked alphabet

$$\Gamma = \bigcup_{0 \leq k \leq N} \Gamma_k$$

we denote by $\Sigma(\Gamma)$ its associated doubly ranked alphabet

$$\Sigma_{m,1}(\Gamma) = \Gamma_m\ (0 \leq m \leq N), \qquad \Sigma_{m,n}(\Gamma) = \emptyset, \text{ otherwise.}$$

Now let $\mathcal{M} = (Q, \delta_{\mathcal{M}}, T)$ be an ordinary tree automaton over Γ where

$$\delta_{\mathcal{M}} \subseteq \bigcup_{k=0}^{N} Q^k \times \Gamma_k \times Q$$

is the move set and $T \subseteq Q$ is the final state set. Then $\mathcal{M}$ can be regarded as a pattern automaton over $\Sigma(\Gamma)$. Once more if $T \in Rat(Q^*)$, the behavior of $\mathcal{M}$ is no longer a tree language but a subset of $mag(\Sigma(\Gamma))$.

Take for instance the tree automaton $\mathcal{M} = (Q = \{q_0, q_1\}, \delta_{\mathcal{M}}, T = \{q_1\})$ over the ranked alphabet $\Gamma = \{f, c\}$ with $rank(f) = 2$, $rank(c) = 0$ and

$$\delta_{\mathcal{M}} = \{((q_0, q_0), f, q_1), ((q_1, q_0), f, q_1), (\varepsilon, c, q_0)\}, \qquad \varepsilon \text{ the empty word.}$$

Its behavior is the tree language $\{t_k \mid k \geq 0, t_0 = c, t_{k+n} = f(t_k, c)\} \subseteq T_\Gamma$. By replacing the final set $T = \{q_1\}$ by $T = \{q_1^3\}$, the obtained behavior is the pattern language

$$\{\begin{pmatrix} t_{k_1} \\ t_{k_2} \\ t_{k_3} \end{pmatrix} \mid k_1, k_2, k_3 \geq 0\}.$$

Actually a notion of a generalized automaton can be defined in any magmoid $M = (M_{m,n})$. It is a system $\mathcal{A} = (M, Q, \theta_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$ where $Q, I_{\mathcal{A}}, T_{\mathcal{A}}$ are as in the definition of a pattern automaton and $\theta_{\mathcal{A}}$ is a finite set of triples (u, a, v) with $a \in M_{|u|,|v|}$. We define the sets $Comp_M(\mathcal{A})$, $Loc_M(\mathcal{A})$ exactly as $Comp(\mathcal{A})$, $Loc(\mathcal{A})$; they are both subsets of $mag(\theta_{\mathcal{A}})$ where $\theta_{\mathcal{A}}$ is considered as a doubly ranked alphabet

$$(\theta_{\mathcal{A}})_{m,n} = \{(u, a, v) \mid |u| = m, |v| = n\}.$$

The behavior of $\mathcal{A}$ is then $|\mathcal{A}| = pr_M(Loc_M(\mathcal{A}))$ where $pr_M : mag(\theta_{\mathcal{A}}) \to M$ is the magmoid morphism defined by $pr_M(u, a, v) = a$. A generalized tree automaton is a system $\mathcal{A} = (Q, \delta_{\mathcal{A}}, T_{\mathcal{A}})$ where $T_{\mathcal{A}} \subseteq Q$ is the final state set and the move set $\delta_{\mathcal{A}}$ consists of triples $((q_1, \ldots, q_n), t, q)$ with $t \in T_\Gamma(X_n)$. Such a system is an automaton over the magmoid $T_\Gamma(X)$ and of course all ordinary tree automata are generalized automata in the above sense.

The class of tree languages recognized by generalized automata properly contains the class of tree languages recognized by ordinary tree automata as it is shown below

Example 3.1 Let $\Gamma = \{f, c\}$, $rank(f) = 2$, $rank(c) = 0$ and consider the simplest generalized tree automaton

$$\mathcal{M} = (\{q\}, \delta_{\mathcal{A}}, \{q\}) \text{ with } \delta_{\mathcal{M}} = \{(q, f(x, x), q), (\varepsilon, c, q)\}.$$

Its behavior is the non recognizable language $\{t_k \mid k \geq 0, t_0 = c, t_{k+1} = f(t_k, t_k)\}$.

Automata admitting ε-moves are called asynchronous. The consideration of such automata does not augment the k-bounded recognition power.

More precisely, an *asynchronous pattern automaton* is a 5-tuple

$$\mathcal{A} = (\Sigma, Q, \mu_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$$

where $\Sigma, Q, I_{\mathcal{A}}, T_{\mathcal{A}}$ are as in the definition of a pattern automaton and $\mu_{\mathcal{A}}$ is a finite set of moves, that is triples either of the form (u, σ, v) $(\sigma \in \Sigma_{m,n}, u \in Q^m, v \in Q^n)$, or of the form (u, ε_m, v) $(u, v \in Q^m)$, where ε_m denotes the m-th empty pattern. In order to define the behavior of such an automaton we first consider the free semimagmoid generated by $\mu_{\mathcal{A}}$, then we construct the sets $Comp(\mathcal{A})$, $Loc(\mathcal{A}) \subseteq smag(\mu_{\mathcal{A}})$, as in the case of pattern automata and finally we take the projection

$$p_\Sigma : smag(\mu_{\mathcal{A}}) \to mag(\Sigma), \qquad p_\Sigma(u,\sigma,v) = \sigma,\ p_\Sigma(u,\varepsilon_m,v) = \varepsilon_m.$$

It holds $|\mathcal{A}| = p_\Sigma(Loc(\mathcal{A}))$. Obviously, an ordinary pattern automaton can be viewed as an asynchronous automaton. Moreover,

THEOREM 3.2 *If $\mathcal{A} = (\Sigma, Q, \theta_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$ is an asynchronous automaton, then $|\mathcal{A}|^{(k)}$ is a recognizable k-bounded pattern language.*

Proof. We consider the pattern automaton $\mathcal{A}' = (\Sigma, Q, \theta_{\mathcal{A}'}, I_{\mathcal{A}}, T_{\mathcal{A}})$ where $\theta_{\mathcal{A}'}$ is defined as follows: for all $u \in Q^m$, $v \in Q^n$, we have $(u,v) \in \theta_{\mathcal{A}'}(\sigma)$, $\sigma \in \Sigma_{m,n}$, if and only if, there exist $u_i \in Q^m$, $v_j \in Q^n$, $1 \le i \le t$, $1 \le j \le s$, such that $(u, \varepsilon_m, u_1), \ldots, (u_{t-1}, \varepsilon_m, u_t) \in Comp(\mathcal{A})$, $(u_t, \sigma, v_1) \in \mu_{\mathcal{A}}$ and $(v_1, \varepsilon_n, v_2), \ldots, (v_s, \varepsilon_n, v) \in Comp(\mathcal{A})$. By construction

$$|\mathcal{A}'| - \{\varepsilon_m \mid m \in \mathbb{N}\} = |\mathcal{A}| - \{\varepsilon_m \mid m \in \mathbb{N}\}.$$

In the case that $\varepsilon_m \in |\mathcal{A}|$ and $\varepsilon_m \notin |\mathcal{A}'|$, $0 \le m \le k$, we add a new state q' in the set Q and the word $q' \cdots q'$ (m times) in the sets $I_{\mathcal{A}}, T_{\mathcal{A}}$. Notice that the restriction $m \le k$ is applied because we are only interested in the k-bounded patterns of $|\mathcal{A}|$.

□

4 Closure Properties

Next normalization result is due to Bossut, Dauchet and Warin:

THEOREM 4.1 (CF. [5]) *Any pattern automaton $\mathcal{A} = (\Sigma, Q, \theta_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$ is equivalent to a pattern automaton $\mathcal{A}' = (\Sigma, Q', \theta_{\mathcal{A}'}, I_{\mathcal{A}'}, T_{\mathcal{A}'})$ with either $I_{\mathcal{A}'}$ or $T_{\mathcal{A}'}$ (but not both) of the form q^*, for $q \in Q'$.*

Now, if the languages $L, L' \subseteq mag(\Sigma)$ are recognized by the automata

$$\mathcal{A}_1 = (\Sigma, Q_1, \theta_{\mathcal{A}_1}, I_{\mathcal{A}_1}, q_+^*) \quad \text{and} \quad \mathcal{A}_2 = (\Sigma, Q_2, \theta_{\mathcal{A}_2}, q_0^*, T_{\mathcal{A}_2})$$

then the automaton $\mathcal{A}$ resulting by merging q_+ with q_0 recognizes the language $L_1 \circ L_2$. Therefore,

PROPOSITION 4.2 (CF. [5]) *The class $Rec(\Sigma)$ is closed under union, intersection, $\circ$-product and $\circ$-star.*

COROLLARY 4.3 *The class $Rec^{(k)}(\Sigma)$ is closed under union and intersection.*

Proof. The proof is obtained by the equalities:

$$(L \cap mag^{(k)}(\Sigma)) \cup (L' \cap mag^{(k)}(\Sigma)) = (L \cup L') \cap mag^{(k)}(\Sigma)$$

and

$$(L \cap mag^{(k)}(\Sigma)) \cap (L' \cap mag^{(k)}(\Sigma)) = (L \cap L') \cap mag^{(k)}(\Sigma).$$

□

It should be pointed out that the technique of [5] cannot be applied to show that the full star of L_1 is also recognizable. Since closure under full star is a crucial point in the theory of pattern languages we shall restrict our notion of a pattern automaton in order to capture this closure property.

A *restricted pattern automaton* is a pattern automaton of the form $\mathcal{A} = (\Sigma, Q, \theta_{\mathcal{A}}, q_0^*, q_+^*)$, $q_0, q_+ \in Q$. The class of the behaviors of all restricted pattern automata over Σ is denoted $RestRec(\Sigma)$.

THEOREM 4.4 *The class $RestRec(\Sigma)$ is closed under union, intersection, $\circ$-product, $\Box$-product and full star.*

THEOREM 4.5 *Let $h : mag(\Sigma) \to mag(\Sigma')$ be a pattern homomorphism*

i) *$L' \in Rec(\Sigma')$ implies $h^{-1}(L') \in Rec(\Sigma)$;*

ii) *if h is non erasing then $L \in Rec(\Sigma)$ implies $h(L) \in Rec(\Sigma')$;*

iii) *if h is k-bounded then $L \in Rec^{(k)}(\Sigma)$ implies $h(L) \in Rec^{(k)}(\Sigma')$;*

iv) *if h is strictly alphabetic then $L' \in Rec^{(k)}(\Sigma')$ implies $h^{-1}(L') \in Rec^{(k)}(\Sigma)$.*

Proof. i) Consider an automaton

$$\mathcal{A}' = (\Sigma', Q', \theta_{\mathcal{A}'}, I_{\mathcal{A}'}, T_{\mathcal{A}'})$$

with $\theta_{\mathcal{A}'} : \Sigma' \to Rel(Q')$. We construct the automaton $\mathcal{A} = (\Sigma, Q, \theta_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$ where $Q = Q'$, $I_{\mathcal{A}} = I_{\mathcal{A}'}$, $T_{\mathcal{A}} = T_{\mathcal{A}'}$ and $\theta_{\mathcal{A}}$ is the composition $\bar{\theta}_{\mathcal{A}'} \circ h$ where $\bar{\theta}_{\mathcal{A}'} : mag(\Sigma') \to Rel(Q')$ is the magmoid morphism granted by Theorem 2.3. From the commutativity of the triangle

$$\begin{array}{ccc} mag(\Sigma) & \xrightarrow{\;h\;} & mag(\Sigma') \\ & \bar{\theta}_{\mathcal{A}} \searrow \quad \swarrow \bar{\theta}_{\mathcal{A}'} & \\ & Rel(Q') & \end{array}$$

we get that $|\mathcal{A}| = h^{-1}(|\mathcal{A}'|)$.

ii) Before proving this assertion we need some auxiliary matter. For any $\mu = (u, p, v)$, $p \in mag_{m,n}(\Sigma')$ and $u = q_1 \cdots q_m$, $v = s_1 \cdots s_n$ (with q_i, s_i taken from a fixed set Q) we shall inductively associate two sets P_μ and Q_μ as follows:

- if $p = \sigma' \in \Sigma'$, then we introduce new states $q_1', \ldots, q_m', s_1', \ldots, s_n'$ and we set

$$P_\mu = \{(u', \sigma', v')\}, \quad u' = q_1' \cdots q_m',\ v' = s_1' \cdots s_n'$$

  and $Q_\mu = \{q_1', \ldots, q_m', s_1', \ldots, s_n'\}$;

- if $p = p_1 p_2$ with $p_1 \in mag_{m,k}(\Sigma')$, $p_2 \in mag_{k,n}(\Sigma')$ we introduce new states $q_1', \ldots, q_m'$, $r_1, \ldots, r_k$, $s_1', \ldots, s_n'$ and we set

$$P_\mu = \{(u', p_1, r_1 \cdots r_k), (r_1 \cdots r_k, p_2, v')\}$$

  and $Q_\mu = \{q_1', \ldots, q_m', r_1, \ldots, r_k, s_1', \ldots, s_n'\}$;

- if $p = \binom{p_1}{p_2}$ with $p_i \in mag_{m_i,n_i}(\Sigma')$, $m_1 + m_2 = m$, $n_1 + n_2 = n$ we introduce new states $q'_1, \ldots, q'_m, s'_1, \ldots, s'_n$ and set

$$P_\mu = \{(q'_1 \cdots q'_{m_1}, p_1, s'_1 \cdots s'_{n_1}), (q'_{m_1+1} \cdots q'_m, p_2, s'_{n_1+1} \cdots s'_n)\}$$

and $Q_\mu = \{q'_1, \ldots, q'_m, s'_1, \ldots, s'_n\}$.

Now let $\mathcal{A} = (\Sigma, Q, \theta_\mathcal{A}, I_\mathcal{A}, T_\mathcal{A})$ be a pattern automaton. We construct an asynchronous pattern automaton $\mathcal{A} = (\Sigma', Q', \theta_{\mathcal{A}'}, I_{\mathcal{A}'}, T_{\mathcal{A}'})$ as follows:

- $Q' = \bigcup Q_\mu$, the union running over all triples $\mu = (u, h(\sigma), v)$, with $(u, \sigma, v) \in \theta_\mathcal{A}$,
- $\theta_{\mathcal{A}'} = \bigcup P_\mu$, the union running over all μ as above,
- $I_{\mathcal{A}'} = \phi(I_\mathcal{A})$, $T_{\mathcal{A}'} = \phi(T_\mathcal{A})$, where $\phi : Q^* \to Q'^*$ is the unique monoid morphism extending the function $q \mapsto q'$.

Clearly, both $I_{\mathcal{A}'}, T_{\mathcal{A}'}$ are rational sets. By construction, $|\mathcal{A}'| = h(|\mathcal{A}|)$.

iii) Starting from an automaton $\mathcal{A}$, we construct an automaton $\mathcal{A}'$ exactly as in item ii) with the addition that: if $h(\sigma) = \varepsilon_n$ ($\sigma \in \Sigma_{n,n}$, ε_n the n-th empty pattern), then from the transition $(q_1 \cdots q_n, \sigma, s_1 \cdots s_n)$ we construct the ε-transition $(\bar{q}_1 \cdots \bar{q}_m, \varepsilon_n, \bar{s}_1 \cdots \bar{s}_n)$, where $\bar{q}_i, \bar{s}_i$ are new states as in ii). The automaton $\mathcal{A}'$ is asynchronous and the result comes by Theorem 3.2. □

iv) Straightforward.

THEOREM 4.6 *The class of k-bounded recognizable pattern languages is closed under k-bounded full star.*

Proof. Given a k-bounded recognizable pattern language L, let

$$\mathcal{A} = (\Sigma, Q, \theta_\mathcal{A}, I_\mathcal{A}, T_\mathcal{A})$$

be a pattern automaton such that $|\mathcal{A}|^{(k)} = L$. Consider now the asynchronous pattern automaton

$$\mathcal{A}' = (\Sigma, Q, \theta_{\mathcal{A}'}, I^*_\mathcal{A}, T^*_\mathcal{A})$$

with $\theta_{\mathcal{A}'} = \theta_\mathcal{A} \cup \mathcal{E}_{\mathcal{A}'}$, where the set $\mathcal{E}_{\mathcal{A}'}$ is defined as follows: if $w \in (T^*_\mathcal{A})^{(k)}$, $|w| = n$, then $(w, \varepsilon_n, u) \in \mathcal{E}_{\mathcal{A}'}$ for all $u \in (I^*_\mathcal{A})^{(k)}$ and $|u| = n$. By Theorem 3.2 we obtain that $|\mathcal{A}'|^{(k)}$ is a recognizable k-bounded pattern language. Furthermore, it holds:

$$|\mathcal{A}'|^{(k)} = L^{\bullet_k}.$$

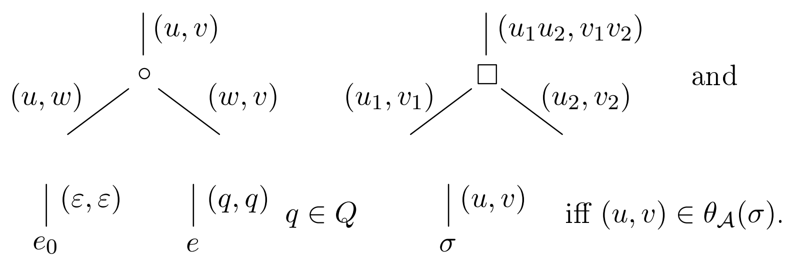
□

5 Decidability Results

Next let $\mathcal{A} = (\Sigma, Q, \delta_\mathcal{A}, I_\mathcal{A}, T_\mathcal{A})$ be a pattern automaton and consider the set of sorts

$$S = \bigcup_{m,n\in\mathbb{N}} Q^m \times Q^n.$$

An S-sorted alphabet $\Gamma(\mathcal{A})$ can be attached to $\mathcal{A}$ consisting of all symbols of the form



The canonical morphism Φ from $\Gamma(\mathcal{A})$ to $\Gamma(\Sigma)$ is obtained via the sort function $S' \to \mathbb{N}\times\mathbb{N}$ which sends the sort (u,v) to the sort $(|u|,|v|)$. We denote by

$$h_\mathcal{A} : T_{\Gamma(\mathcal{A})} \to mag(\Sigma)$$

the composition

$$T_{\Gamma(\mathcal{A})} \xrightarrow{\Phi} T_{\Gamma(\Sigma)} \xrightarrow{h_\Sigma} mag(\Sigma).$$

The following important result is obtained.

THEOREM 5.1 *Given $k \geq 1$, to any pattern automaton $\mathcal{A} = (\Sigma, Q, \theta_\mathcal{A}, I_\mathcal{A}, T_\mathcal{A})$ we can associate a positive integer $c_\mathcal{A}^{(k)}$ so that, for any $p \in |\mathcal{A}|^{(k)}$, we can effectively construct a pattern $\bar{p}$ with $height(\bar{p}) \leq c_\mathcal{A}^{(k)}$ such that $\bar{p} \in |\mathcal{A}|^{(k)}$.*

Proof. We put

$$c_\mathcal{A}^{(k)} = card(\bigcup_{m,n\leq k} Q^m \times Q^n).$$

Now, if $t \in T_{\Gamma(\mathcal{A})}$ has the properties

$$\Phi(t) \in T^{(k)}_{\Gamma(\Sigma)}, \quad h_\mathcal{A}(t) = p$$

and $height(t) > c_\mathcal{A}^{(k)}$, there is a path (d) of t having length $> c_\mathcal{A}^{(k)}$, which implies that two state pairs in (d), say (u_i, v_i) and (u_j, v_j), must be equal:

$$(u_i, v_i) = (u_j, v_j), \qquad i < j.$$

By dropping the part of t between these two equal pairs we get a tree $t' \in T_{\Gamma(\mathcal{A})}$ such that

$$height(t') \leq height(t) \text{ and } h_\mathcal{A}(t') \in |\mathcal{A}|^{(k)}.$$

By repeating, if necessary, the above procedure, we arrive to a tree $\bar{t} \in T_{\Gamma(\mathcal{A})}$, such that,

$$height(\bar{t}) \leq c_{\mathcal{A}}^{(k)} \text{ and } h_{\mathcal{A}}(\bar{t}) = \bar{p} \in |\mathcal{A}|^{(k)}$$

as wanted. □

COROLLARY 5.2 *We can decide in polynomial time whether the k-bounded behavior of a pattern automaton $\mathcal{A}$ is empty or not.*

In order to get a pumping lemma in this framework we need to speak of frames. Let $\xi_{m,n}$ be a new symbol with rank (m,n) and denote by $FR_{m,n}^{\alpha,\beta}(\Sigma)$ the subset of $mag_{\alpha,\beta}(\Sigma \cup \{\xi_{m,n}\})$ with just one occurence of $\xi_{m,n}$. The elements of $FR_{m,n}^{\alpha,\beta}(\Sigma)$ are called frames with exterior rank (α,β) and interior rank (m,n). Notice that $\xi_{m,n} \in FR_{m,n}^{m,n}(\Sigma)$. The set $FR_{m,n}^{\alpha,\beta}(\Sigma)$ acts on $mag_{m,n}(\Sigma)$ via substitution at $\xi_{m,n}$: for $f \in FR_{m,n}^{\alpha,\beta}(\Sigma)$ and $p \in mag_{m,n}(\Sigma)$,

$$f \cdot p = f[p/\xi_{m,n}].$$

The argument in the proof of Theorem 5.1 can be used to show the next pumping result.

THEOREM 5.3 *To any k-bounded recognizable pattern language $L \subseteq mag^{(k)}(\Sigma)$ we can attach a positive integer c such that for any pattern $p \in L_{m,n}$ we can determine k-bounded frames $f_1 \in FR_{\alpha,\beta}^{m,n}(\Sigma)$, $f_2 \in FR_{\alpha,\beta}^{\alpha,\beta}(\Sigma)$ and $q \in mag_{\alpha,\beta}^{(k)}(\Sigma)$ so that p admits the factorization $p = f_1 \cdot f_2 \cdot q$, where f_2 is an nontrivial frame i.e., $f_2 \neq \xi_{\alpha,\beta}$ and $height(q) \leq c$. Moreover, it holds*

$$f_1 \cdot f_2^{\lambda} \cdot q \in L$$

for all $\lambda \in \mathbb{N}$.

6 Context Free Pattern Languages

In this section we examine context free pattern grammars which are identical with the well known hyperedge replacement grammars (cf. [11, 12, 18, 19]) by viewing a pattern as a planar directed acyclic graph. It is important to point out that the class of such languages is organized into a cone with respect to recognizable pattern transductions as we shall see in Section 9.

A *context free patern grammar* is a triple $\mathcal{G} = (\Sigma, X, \mathcal{R})$, where Σ is the terminal alphabet, $X = \{x_1, \ldots, x_n\}$ with $rank(x_i) = (m_i, n_i)$ is the nonterminal alphabet and $\mathcal{R}$ is a finite set of rules of the form $x_i \to p$, with $p \in mag_{m_i,n_i}(\Sigma \cup X)$. For patterns $p, p' \in mag_{\alpha,\beta}(\Sigma \cup X)$ we write $p \Rightarrow p'$ whenever there is a frame $f \in FR_{m_i,n_i}^{\alpha,\beta}(\Sigma)$ and a rule $x \to \bar{p}$, so that

$$p = f \cdot x, \quad p' = f \cdot \bar{p}.$$

The pattern language generated by $\mathcal{G}$, starting from x_i, is given by

$$L(\mathcal{G}, x_i) = \{p \mid p \in mag_{m_i,n_i}(\Sigma), x_i \overset{*}{\Rightarrow} p\}.$$

Pattern languages coming in this way are called *context free*. We denote by $CPL(\Sigma)$ the class of all context free patern languages over Σ.

EXAMPLE 6.1 We shall display a context free graph grammar generating the set L_n of all graphs representing the linear functions with n variables. Consider the alphabet $\Sigma = \{0, succ, mult, sum_n\}$ with $rank(0) = (1,0)$, $rank(succ) = (1,1)$, $rank(mult) = (2,1)$, $rank(sum_n) = (n,1)$, the variables $x_{0,1}, x_{1,1}, x_{n,1}$ with ranks $(0,1), (1,1), (n,1)$ respectively and the grammar $\mathcal{G}$:

$$x_{0,1} \to 0, \quad x_{0,1} \to x_{0,1} \circ succ, \quad x_{1,1} \to (x_{0,1} \,\square\, E) \circ mult,$$

$$x_{n,1} \to (x_{1,1} \,\square \cdots \square\, x_{1,1}) \circ sum_n.$$

The first two rules generate the set of graphs

$$G_\alpha = 0 \circ succ \circ \cdots \circ succ \qquad (\alpha \text{ times}, \alpha \in \mathbb{N})$$

and together with the third rule generate the graphs

$$G_{\alpha x} = (G_\alpha \,\square\, E) \circ mult.$$

Finally with the use of the last rule we take the graphs

$$G_{\alpha_1 x_1 + \cdots + \alpha_n x_n} = (G_{\alpha_1 x_1} \,\square \cdots \square\, G_{\alpha_n x_n}) \circ sum_n.$$

If we interpret the symbol 0 as the zero natural number, $succ$ as the successor function in $\mathbb{N}$, E as the identity function $\mathbb{N} \to \mathbb{N}$, $mult : \mathbb{N}^2 \to \mathbb{N}$ as the usual multiplication and $sum_n : \mathbb{N}^n \to \mathbb{N}$ as the usual n argument sum, then the graph $G_{\alpha_1 x_1 + \cdots + \alpha_n x_n}$ represents the linear function $f : \mathbb{N}^n \to \mathbb{N}$ with

$$f(x_1, \ldots, x_n) = \alpha_1 x_1 + \cdots + \alpha_n x_n.$$

PROPOSITION 6.2 (CHOMSKY LIKE NORMAL FORM) *Given a context free pattern grammar $\mathcal{G}$ we can effectively construct an equivalent context free pattern grammar $\mathcal{G}'$ whose rules are of the form*

$$x \to y\, z, \quad x \to \begin{pmatrix} y \\ z \end{pmatrix}, \quad x \to e, \quad x \to e_0, \quad x \to \sigma \ \ (\sigma \in \Sigma).$$

Proof. The proof is obtained from [13], Theorem 7. □

An immediate consequence of the above proposition is the next decidability result.

PROPOSITION 6.3 *For a context free pattern language L we can decide whether $L = \emptyset$ or not.*

Fact. Every context free pattern language $L \subseteq mag_{m,n}(\Sigma)$ is k-bounded for some $k \geq 0$. Indeed, assume that L is generated by the context free pattern grammar in Chomsky normal form $\mathcal{G} = (\Sigma, X, \mathcal{R})$, then the bound k is the maximum of the ranks of all the variables in X.

THEOREM 6.4 *The intersection of a context free with a recognizable (resp. k-bounded recognizable) pattern language is again context free.*

Proof. Consider a pattern automaton $\mathcal{A} = (\Sigma, Q, \theta_\mathcal{A}, I_\mathcal{A}, T_\mathcal{A})$, and a context free pattern grammar $\mathcal{G} = (\Sigma, X, \mathcal{R})$ in Chomsky normal form, i.e., its rules are of the form

$$x \to yz, \ \ x \to \begin{pmatrix} y \\ z \end{pmatrix}, \ \ x \to e, \ \ x \to e_0, \ \ x \to \sigma \ (\sigma \in \Sigma).$$

We introduce the new set of variables X':

$$(u, x, v), \ \ u \in Q^m, v \in Q^n$$

and we consider the pattern grammar $\mathcal{G}' = (\Sigma, X', \mathcal{R}')$ whose rules are of the form:

- $(u, x, v) \to (u, x_1, w)(w, x_2, v)$, whenever $x \to x_1 x_2$ is in $\mathcal{R}$;
- $(u, x, v) \to \begin{pmatrix} (u_1, x_1, v_1) \\ (u_2, x_2, v_2) \end{pmatrix}$, whenever $x \to \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ is in $\mathcal{R}$ and $u = u_1 u_2$, $v = v_1 v_2$;
- $(u, x, v) \to \sigma$, whenever $x \to \sigma$ is in $\mathcal{R}$ and $(u, v) \in \delta_{\mathcal{A}}(\sigma)$, $\sigma \in \Sigma$;
- $(q, x, q) \to e$, whenever $x \to e$ is in $\mathcal{R}$;
- $(\varepsilon, x, \varepsilon) \to e_0$, whenever $x \to e_0$ is in $\mathcal{R}$.

Then for every $p \in mag_{m,n}(\Sigma)$ and $(u, x, v) \in X'$, with $u \in Q^m$, $v \in Q^n$, we get (by induction) that

$$(u, x, v) \overset{*}{\Rightarrow} p \qquad \text{iff} \qquad x \overset{*}{\Rightarrow} p \ \text{ and } \ (u, v) \in \bar{\delta}_{\mathcal{A}}(p).$$

Hence for all $x \in X$ with $rank(x) = (m, n)$, it holds

$$\bigcup_{u \in I_{\mathcal{A}}^{(m)}, v \in T_{\mathcal{A}}^{(n)}} L(\mathcal{G}', (u, x, v)) = L(\mathcal{G}, x) \cap |\mathcal{A}|_{m,n}.$$

The k-bounded case is treated analogously. □

As we have seen, for any context free pattern grammar $\mathcal{G} = (\Sigma, X, \mathcal{R})$, the language $L(\mathcal{G}, x) \subseteq mag(\Sigma)$ $(x \in X)$, is k_0-bounded for a fixed $k_0 \geq 1$. Now given $k < k_0$, the language $L^{(k)}(\mathcal{G}, x)$ formed by all k-bounded patterns that can be generated by $\mathcal{G}$ starting from x, is also context free. Indeed, it holds

$$L^{(k)}(\mathcal{G}, x) = L(\mathcal{G}, x) \cap mag^{(k)}(\Sigma)$$

and $mag^{(k)}(\Sigma)$ is the k-bounded behavior of the one state automaton $\mathcal{A}$ with moves

$$(q^m, \sigma, q^n), \quad \sigma \in \Sigma_{m,n}, \ m, n \geq 1,$$

whose initial and final state languages are both q^*. Hence,

PROPOSITION 6.5 *If the language $L \subseteq mag(\Sigma)$ is context free, then so is $L^{(k)}$ for all $k \geq 1$.*

A pattern $p \in mag(\Sigma)$ is called *connected* whenever it cannot be factorized as

$$p = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}$$

with $p_1, p_2 \neq \varepsilon_0$. In [5] it was shown that the set $Con(\Sigma)$ of all connected patterns over Σ is a recognizable pattern language. Thus, using the previous theorem, we get that

COROLLARY 6.6 *For any context free pattern language $L \subseteq mag(\Sigma)$, its connected part*

$$Con(L) = L \cap Con(\Sigma)$$

is context free as well.

Consequently, we can decide whether a context free patter language contains at least one connected pattern.

Proposition 6.7 *The class CPL is closed under pattern homomorphisms and inverse strictly alphabetic homomorphisms.*

Proof. Consider a context free pattern grammar

$$\mathcal{G} = (\Sigma, X, \mathcal{R})$$

and the homomorphism $h : mag(\Sigma) \to mag(\Sigma')$, then the grammar

$$h(\mathcal{G}) = (\Sigma', X, h(\mathcal{R})),$$

with $h(\mathcal{R}) = \{x \to h(p) \mid x \to p \in \mathcal{R}\}$, generates the language $h(L(\mathcal{G}, x))$, $x \in X$.

Analogously, if $\varphi : mag(\Delta) \to mag(\Sigma)$ is a strictly alphabetic homomorphism, then the grammar

$$\varphi^{-1}(\mathcal{G}) = (\Delta, X, \varphi^{-1}(\mathcal{R}))$$

with

$$\varphi^{-1}(\mathcal{R}) = \{x \to q \mid q \in mag(\Delta),\ x \to \varphi(q) \in \mathcal{R}\}$$

generates $\varphi^{-1}(L(\mathcal{G}, x))$, $x \in X$. □

Proposition 6.8 *Let $L \subseteq GR_{\alpha,\beta}(\Sigma)$ be an infinite context free language. Then every graph $G \in L$ with large enough size can be factorized as*

$$G = F_1 \cdot F_2 \cdot G'$$

with $F_1 \in FR^{\alpha,\beta}_{m,n}(\Sigma)$, $F_2 \in FR^{m,n}_{m,n}$, $G' \in GR_{m,n}(\Sigma)$, $F_2 \neq \xi_{m,n}$, so that

$$F_1 \cdot F_2^k \cdot G' \in L \qquad \text{for all } k \geq 0.$$

Proof. Assume that L is generated by a context free graph grammar $G = (\Sigma, X, \mathcal{R})$ in Chomsky normal form. Since $size(G)$ is large enough there is a derivation of G in which the same rule $x \to \bar{G}$ of $\mathcal{R}$ is used twice:

$$x_i \overset{*}{\Rightarrow} F_1 \cdot x \Rightarrow F_1 \cdot \bar{G} \overset{*}{\Rightarrow} F_1 \cdot F_2 \cdot x \Rightarrow F_1 \cdot F_2 \cdot \bar{G} \overset{*}{\Rightarrow} F_1 \cdot F_2 \cdot G' = G.$$

The result follows immediately. □

Example 6.9 The language $L = \{G \,\square\, G \mid G \in GR_{m,n}(\Sigma)\}$, is not context free. Indeed, assume that L is context free, then according to the above proposition, given a graph $G \,\square\, G$, $G \in GR_{\alpha,\beta}(\Sigma)$, there exist $F_1 \in FR^{\alpha,\beta}_{m,n}(\Sigma)$, $F_2 \in FR^{m,n}_{m,n}$, $G' \in GR_{m,n}(\Sigma)$, $F_2 \neq \xi_{m,n}$, so that

$$G \,\square\, G = F_1 \cdot F_2 \cdot G'.$$

Hence $F_2 \cdot G'$ belongs either at the first or at the second part of the graph $G \,\square\, G$ (but not at both since the graph is disjoint). Moreover, for all $k \geq 0$ we have

$$F_1 \cdot F_2^k \cdot G' \in L,$$

which is a contradiction because, for $k \geq 2$, the above graph doesn't consist of two identical parts.

Since every context free graph language L has a specified rank i.e., $L \subseteq GR_{m,n}(\Sigma)$ for some $m, n \geq 0$, whereas every recognizable graph language generally expands to any rank, in order to compare these two classes we have to restrict ourselves to recognizable graph languages of a certain rank.

Proposition 6.10 *The classes of context free and recognizable graph languages are incomparable.*

Proof. One direction is obtained from the fact that every singleton language consisting of a non-discrete graph is manifestly context free but not recognizable. Indeed let $F \in GR_{m,n}(\Sigma)$ be a non-discrete graph and $\mathcal{A}$ an automaton recognizing the language $L = \{F\}$. Then we observe that $|\mathcal{A}|$ should also contain the graph

$$I'_{m,2} \circ (F \,\square\, F) \circ I'_{2,n}$$

where $I'_{k,2}$ and $I'_{2,k}$, $k \geq 0$, are the *identificator graphs* (introduced in Sect. 6.1 of [7]), i.e., the discrete graphs that have the set of nodes $\{x_1, \ldots, x_k\}$ and the following begin and end sequences respectively: $x_1 \cdots x_k$ and $x_1 \cdots x_k x_1 \cdots x_k$, and $x_1 \cdots x_k x_1 \cdots x_k$ and $x_1 \cdots x_k$.

In the opposite direction the language $GR_{m,n}(\Sigma)$ is manifestly recognizable but fails to be context free. Indeed, if F is the complete graph with n nodes $\{v_1, \ldots, v_n\}$ then the complete graph with $n+1$ nodes is obtained by adding a new node v_{n+1} and n edges from v_{n+1} to each one the nodes $\{v_1, \ldots, v_n\}$. Thus if a context free graph grammar would generate $GR_{m,n}(\Sigma)$ then necessarily, in order to generate all the complete graphs, it should contain variables of arbitrary large rank. □

7 Finite State Pattern Languages

The objective of this section is to study k-bounded pattern languages admitting a finite number of derivatives. This class of languages is very important because the equality problem is proved to be solvable.

Consider a pattern automaton $\mathcal{A} = (\Sigma, Q, \theta_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$ and the associated many sorted alphabet $\Gamma(\mathcal{A})$ as introduced at the beginning of Section 5. We have a canonical mapping $\pi_{\mathcal{A}} : T_{\Gamma(\mathcal{A})} \to mag(\Sigma \times \mathbf{Q})$ inductively defined by the following rules

$$\begin{array}{c}(u,v)\\ |\\ \sigma\end{array} \overset{\pi_{\mathcal{A}}}{\longmapsto} (u,\sigma,v), \quad (u,v) \in \theta_{\mathcal{A}}(\sigma), \quad \begin{array}{c}(q,q)\\ |\\ e\end{array} \overset{\pi_{\mathcal{A}}}{\longmapsto} (q,e,q), \quad \begin{array}{c}(\varepsilon,\varepsilon)\\ |\\ e_0\end{array} \overset{\pi_{\mathcal{A}}}{\longmapsto} (\varepsilon,e_0,\varepsilon),$$

$$\begin{array}{ccc} & (u,v) & \\ & | & \\ (u,w) & \circ & (w,v) \\ \swarrow & & \searrow \\ t_1 & & t_2 \end{array} \overset{\pi_{\mathcal{A}}}{\longmapsto} \pi_{\mathcal{A}}(t_1)\, \pi_{\mathcal{A}}(t_2), \qquad t_1 \in T^{(u,w)}_{\Gamma(\mathcal{A})},\ t_2 \in T^{(w,v)}_{\Gamma(\mathcal{A})},$$

$$\begin{array}{ccc} & (u_1u_2, v_1v_2) & \\ & | & \\ (u_1,v_1) & \square & (u_2,v_2) \\ \swarrow & & \searrow \\ t_1 & & t_2 \end{array} \overset{\pi_{\mathcal{A}}}{\longmapsto} \begin{pmatrix}\pi_{\mathcal{A}}(t_1)\\ \pi_{\mathcal{A}}(t_2)\end{pmatrix}, \qquad t_1 \in T^{(u_1,v_1)}_{\Gamma(\mathcal{A})},\ t_2 \in T^{(u_2,v_2)}_{\Gamma(\mathcal{A})}.$$

Next triangle clearly commutes.

$$\begin{array}{ccc} T_{\Gamma(\mathcal{A})} & \xrightarrow{\pi_{\mathcal{A}}} & mag(\Sigma \times \mathbf{Q}) \\ & {\scriptstyle h_{\mathcal{A}}} \searrow & \downarrow {\scriptstyle pr_{\Sigma}} \\ & & mag(\Sigma) \end{array} \qquad (p_1)$$

Recall that a tree $t \in T_{\Gamma(\mathcal{A})}^{(u,v)}$ is said to be k-bounded if all sorts u', v' occurring in t have lengths $\leq k$, i.e., $|u'|, |v'| \leq k$. This means that $\Phi(t) \in T_{\Gamma(\Sigma)}^{(k)}$, where $\Phi : T_{\Gamma(\mathcal{A})} \to T_{\Gamma(\Sigma)}$ is the morphism defined also in Section 5. We denote by $T_{\Gamma(\mathcal{A})}^{(k)}$ the set of all k-bounded trees $t \in T_{\Gamma(\mathcal{A})}^{(u,v)}$, $(u, v \in Q^*)$. Any language $\Lambda \subseteq T_{\Gamma(\mathcal{A})}^{(k)}$ is called k-bounded.

In order to construct frames in this many sorted case we take the auxiliary alphabet $\Xi = \{\xi_{m,n} \mid m, n \in \mathbb{N}\}$ and we add sorts over Q i.e., we consider the alphabet

$$\Xi(Q) = \{(u, \xi_{m,n}, v) \mid u \in Q^m, v \in Q^n, m, n \in \mathbb{N}\}.$$

There results a canonical mapping

$$\chi_{\mathcal{A}} : (P_{\Gamma(\mathcal{A}) \cup \Xi(Q)})_{(u',v')}^{(u,v)} \to FR_{(|u'|,|v'|)}^{(|u|,|v|)}(\Sigma)$$

which is defined exactly as $\pi_{\mathcal{A}}$ above with the additional rule

$$\begin{array}{c} (u,v) \\ | \\ (u, \xi_{m,n}, v) \end{array} \quad \overset{\chi_{\mathcal{A}}}{\longmapsto} \quad \xi_{m,n}.$$

Fact. For all $\tau \in (P_{\Gamma(\mathcal{A}) \cup \Xi(Q)})_{(u',v')}^{(u,v)}$ and $t \in T_{\Gamma(\mathcal{A})}^{(u',v')}$ we have

$$\pi_{\mathcal{A}}(\tau t) = \chi_{\mathcal{A}}(\tau) \cdot \pi_{\mathcal{A}}(t). \qquad (p_2)$$

Furthermore, we denote by $P_{\Gamma(\mathcal{A})}^{(k)}$ the subset of

$$\bigcup_{u,v,u',v'} (P_{\Gamma(\mathcal{A})})_{(u',v')}^{(u,v)}$$

consisting of all pruned trees over the alphabet $(\Gamma(\mathcal{A}))$ in which all the occurring sorts $(\bar{u}, \bar{v})$ satisfy $|\bar{u}|, |\bar{v}| \leq k$.

The *k-bounded derivative* of $\Lambda \subseteq T_{\Gamma(\mathcal{A})}^{(k)}$ at $t \in T_{\Gamma(\mathcal{A})}^{(k)}$ is the set

$$t^{-k}\Lambda = \{\tau \mid \tau \in P_{\Gamma(\mathcal{A})}^{(k)}, \tau t \in \Lambda\}.$$

Furthermore, the k-bounded derivative of the set $L \subseteq mag^{(k)}(\Sigma)$ at $p \in mag^{(k)}(\Sigma)$ is the set of all k-bounded frames f such that $f\,p \in L$, *i.e.*,

$$p^{-k}L = \{f \mid f \in FR^{(k)}(\Sigma), f \cdot p \in L\}.$$

Call a k-bounded language $L \subseteq mag^{(k)}(\Sigma)$ *finite state*, if it has finitely many distinct k-bounded derivatives.

$$card\{p^{-k}L \mid p \in mag^{(k)}(\Sigma)\} < \infty.$$

THEOREM 7.1 *Every k-bounded recognizable pattern language $L \subseteq mag^{(k)}(\Sigma)$ is finite state.*

Proof. For any $p \in mag^{(k)}(\Sigma)$ there is a $t \in T^{(k)}_{\Gamma(\mathcal{A})}$ so that,

$$h_{\mathcal{A}}(t) = p \quad \text{and} \quad h_{\mathcal{A}}(t^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})})) = p^{-k}L.$$

Fact. For k-bounded trees t, t' with $t \in T^{(u,v)}_{\Gamma(\mathcal{A})}$, $t' \in T^{(u',v')}_{\Gamma(\mathcal{A})}$ such that $h_{\mathcal{A}}(t) = h_{\mathcal{A}}(t')$ it holds

$$t^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})}) = t'^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})}) \tag{1}$$

if and only if

$$u = u' \text{ and } v = v'.$$

Indeed, if $t^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})}) \neq \emptyset$ there is a pruned tree $\tau \in (P_{\Gamma(\mathcal{A})})^{(r,s)}_{(u,v)}$ so that

$$\tau t \in (h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})}).$$

Thus if equality (1) holds then $\tau \in t'^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})})$. It follows that $(u', v') = (u, v)$ as wanted.

Now assume that $u = u'$ and $v = v'$ and let $\tau \in t^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})})$ then

$$\tau \cdot t \in h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})} \quad \text{or} \quad h_{\mathcal{A}}(\tau \cdot t) \in L.$$

But we have

$$\begin{aligned} h_{\mathcal{A}}(\tau \cdot t) &\overset{(p_1)}{=} pr_{\Sigma}(\pi_{\mathcal{A}}(\tau \cdot t)) \overset{(p_2)}{=} pr_{\Sigma}(\chi_{\mathcal{A}}(\tau) \cdot \pi_{\mathcal{A}}(t)) = pr_{\Sigma}(\chi_{\mathcal{A}}(\tau)) \cdot pr_{\Sigma}(\pi_{\mathcal{A}}(t)) \\ &\overset{(p_1)}{=} pr_{\Sigma}(\chi_{\mathcal{A}}(\tau)) \cdot h_{\mathcal{A}}(t) \overset{hyp.}{=} pr_{\Sigma}(\chi_{\mathcal{A}}(\tau)) \cdot h_{\mathcal{A}}(t') = \cdots = h_{\mathcal{A}}(\tau \cdot t'). \end{aligned}$$

Hence, $\tau \cdot t' \in h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})}$ i.e., $\tau \in t'^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})})$ and thus

$$t^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})}) \subseteq t'^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})}).$$

The opposite inclusion follows by an analogous argument.

Since there are finitely many (u, v) with $|u|, |v| \leq k$, we get that

$$card\{t^{-k}(h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})})\} < \infty.$$

But the k-bounded derivatives of L are exactly the projections via $h_{\mathcal{A}}$ of the corresponding derivatives of $h^{-1}_{\mathcal{A}}(L) \cap T^{(k)}_{\Gamma(\mathcal{A})}$. The result follows. □

Using a classical argument we obtain the next standard result.

PROPOSITION 7.2 *The finite state pattern languages are closed under the boolean operations of union, intersection and complement.*

THEOREM 7.3 *Every k-bounded finite state language $L \subseteq mag^{(k)}(\Sigma)$ is context free.*

Proof.

Let $p_1^{-k}L, \ldots, p_m^{-k}L$ be the distinct k-bounded derivatives of L and assume that

$$p_1, \ldots, p_\alpha \in L \quad \text{while} \quad p_{\alpha+1}, \ldots, p_m \notin L.$$

We introduce the variables $x_{p_i^{-k}L}$ of rank equal with the rank of p_i $(1 \leq i \leq m)$ and the context free pattern grammar G :

$$x_{p_i^{-k}L} \to x_{p_j^{-k}L} x_{p_m^{-k}L}, \quad \text{iff} \quad (p_j p_m)^{-k} L = p_i^{-k} L,$$

$$x_{p_i^{-k}L} \to \begin{pmatrix} x_{p_j^{-k}L} \\ x_{p_m^{-k}L} \end{pmatrix}, \quad \text{iff} \quad \begin{pmatrix} p_j \\ p_m \end{pmatrix}^{-k} L = p_i^{-k} L,$$

$$x_{p_i^{-k}L} \to a, \quad \text{iff} \quad p_i^{-k} L = a^{-k} L, \quad a \in \Sigma \cup \{e, e_0\}.$$

Then it holds

$$x_{p_i^{-k}L} \overset{*}{\underset{G}{\Rightarrow}} \bar{p} \quad \text{iff} \quad p_i^{-k} L = \bar{p}^{-k} L$$

and thus

$$L = \bigcup_{1 \leq i \leq \alpha} L(G, x_{p_i^{-k}L}).$$

The conclusion follows. □

By combining Theorems 7.1 and 7.3 we obtain the following.

COROLLARY 7.4 *Every k-bounded recognizable pattern language is context free.*

Since the emptiness problem is decidable for context free pattern languages, according to the previous theorem, the problem is also decidable for k-bounded finite state pattern languages. But the last class is closed under boolean operations. Hence for such languages L, L' we have $L \subseteq L'$, if and only if, $L \cap \bar{L}' = \emptyset$, where $\bar{L}$ stands for the set theoretic complement of L.

THEOREM 7.5 *The equality problem is decidable for k-bounded finite state pattern languages. Consequently the equality problem is also decidable for k-bounded recognizable pattern languages.*

8 Application to 0-type Grammars and Word Rewriting Systems

Pattern automata are very closely related with 0-type grammars. Recall that a 0-type grammar is a triple $G = (\Sigma, X, R, w_0)$ where Σ, X are the finite alphabets of terminal and non terminal symbols, $w_0 \in (\Sigma \cup X)^*$ is the axiom, and R is a finite set of rules $u \to v$ with $u, v \in (\Sigma \cup X)^*$. For $w, w' \in (\Sigma \cup X)^*$ we write

$$w \underset{G}{\Rightarrow} w' \quad \text{if} \quad w = w_1 u w_2, \;\; w' = w_1 v w_2 \quad \text{and} \quad u \to v \in R.$$

As usual $\underset{G}{\Rightarrow}^*$ denotes the reflexive and transitive closure of $\underset{G}{\Rightarrow}$ and the language generated by G is

$$L(G) = \{w \mid w \in \Sigma^*, w_0 \underset{G}{\Rightarrow}^* w\}.$$

To any 0-type grammar $G = (\Sigma, X, R, w_0)$ we may associate a doubly ranked alphabet $\Sigma(G) = \{\sigma_{u,v} \mid u \to v \in R, rank(\sigma_{u,v}) = (|u|, |v|)\}$, as well as a pattern automaton

$$\mathcal{A}(G) = (\Sigma(G), \Sigma \cup X, \delta_{\mathcal{A}(G)}, w_0, \Sigma^*)$$

with $\delta_{\mathcal{A}(G)} = \{(u, \sigma_{u,v}, v) \mid u \to v \in R\}$. The set $Loc(\mathcal{A}(G))$ is the *derivations pattern language of* G. Then $L(G)$ is just the image of $Loc(\mathcal{A}(G))$ through the face function

$$\partial_1 : Loc(\mathcal{A}(G)) \to \Sigma^*, \quad \partial_1(Loc(\mathcal{A}(G))) = L(G).$$

If $k \geq 1$, then $L^{(k)}(G)$ is the image of $Loc(\mathcal{A}(G)) \cap mag^{(k)}(\Sigma(G))$ via ∂_1. The following proposition has to be added to the rather poor set of decidability results concerning 0-type grammars.

PROPOSITION 8.1 *Given a* 0*-type grammar* G *and a positive integer* k *we can decide whether or not* $L^{(k)}(G) \neq \emptyset$*, that is whether or not there is a* k*-bounded successful derivation in* G*.*

Proof. Since $Loc^{(k)}(\mathcal{A}(G))$ is a k-bounded recognizable pattern language, it is decidable whether or not $Loc^{(k)}(\mathcal{A}(G)) \neq \emptyset$ (see Corollary 5.2). The result follows from the equality

$$L^{(k)}(G) = \partial_1(Loc^{(k)}(\mathcal{A}(G))).$$

□

Let us now pass to rewriting systems. A *rewriting system* is a pair $\mathcal{R} = (X, R)$ formed by a set X (the carrier set) and a relation $R \subseteq X^* \times X^*$. We often say that $(u, v) \in R$ is a rule of $\mathcal{R}$ and we write $u \to v \in R$. For $w_1, w_2 \in X^*$, we set $w_1 \underset{\mathcal{R}}{\Rightarrow} w_2$ if $w_1 = wuw'$, $w_2 = wvw'$ and $u \to v \in R$. Again $\underset{\mathcal{R}}{\Rightarrow}^*$ stands for the reflexive and transitive closure of $\underset{\mathcal{R}}{\Rightarrow}$. A production in the writing system $\mathcal{R}$

$$w_0 \underset{\mathcal{R}}{\Rightarrow} w_1 \underset{\mathcal{R}}{\Rightarrow} w_2 \underset{\mathcal{R}}{\Rightarrow} \cdots$$

is said to be k*-bounded* if $|w_i| \leq k$ for all $i = 0, 1, 2, \ldots$. We say that $w \in X^*$ is in *normal form* if there is no $w' \in X^*$ such that $w \underset{\mathcal{R}}{\Rightarrow} w'$.

In the sequel we deal with finite rewriting systems $\mathcal{R} = (X, R)$, i.e., $card(R) < \infty$. Let us consider the pattern automaton

$$\mathcal{A}(\mathcal{R}) = (\Sigma(\mathcal{R}), X, \delta_{\mathcal{A}(\mathcal{R})}, X^*, X^*)$$

with $\Sigma(\mathcal{R}) = \{\sigma_{|u|,|v|} \mid u \to v \in R\}$ and

$$\delta_{\mathcal{A}(\mathcal{R})} = \{(u, \sigma_{|u|,|v|}, v) \mid u \to v \in R\}.$$

The behavior $|\mathcal{A}(\mathcal{R})|$ of this automaton is the set of all derivation patterns of the rewriting system $\mathcal{R}$, while its k-bounded behavior $|\mathcal{A}(\mathcal{R})|^{(k)}$, is the set of all k-bounded derivation patterns of $\mathcal{R}$. Furthermore, the relation generated by the system $\mathcal{R}$ is the image of $Loc(\mathcal{A}(\mathcal{R}))$ via the double face function

$$< \partial_0, \partial_1 >: Loc(\mathcal{A}(\mathcal{R})) \to X^* \times X^*, \qquad < \partial_0, \partial_1 > (p) = (\partial_0(p), \partial_1(p)),$$

and the k-bounded relation generated by $\mathcal{R}$ is the image of $Loc^{(k)}(\mathcal{A}(\mathcal{R}))$ again via the same function.

Denoting by $\mathcal{R}^{(k)}(u, v)$ the set of all k-bounded derivation patterns starting from u and ending to v, the confluence condition for k-bounded productions of $\mathcal{R}$ can be formulated as follows: for all $u, v_1, v_2 \in X^*$

$$\mathcal{R}^{(k)}(u, v_1) \neq \emptyset,\ \mathcal{R}^{(k)}(u, v_2) \neq \emptyset \quad \text{implies} \quad \mathcal{R}^{(k)}(v_1, v) \neq \emptyset,\ \mathcal{R}^{(k)}(v_2, v) \neq \emptyset$$

for some $v \in X^*$. Hence, by virtue of Corollary 5.2, we obtain the following result.

PROPOSITION 8.2 *Given a rewriting system $\mathcal{R} = (X, R)$, the confluence problem is decidable for k-bounded productions.*

A rewriting system is said to be *k-bounded terminating* if it admits no endless k-bounded productions.

PROPOSITION 8.3 *For a specified $k \geq 1$, we can decide whether a rewriting system $\mathcal{R} = (X, R)$ is k-bounded terminating.*

Proof. First we determine the finite set

$$NF^{(k)}(\mathcal{R}) = \{w \mid |w| \leq k,\ w \text{ in normal form } \}$$

and then we construct the pattern automaton

$$\mathcal{A}_{ter}(\mathcal{R}) = (\Sigma(\mathcal{R}), X, \delta_{\mathcal{A}(\mathcal{R})}, \bigcup_{p \leq k} X^p, NF^{(k)}(\mathcal{R})).$$

The set of all k-bounded productions from $\bigcup_{p \leq k} X^p$ to $NF^{(k)}(\mathcal{R})$ is just the image of $Loc^{(k)}(\mathcal{A}_{ter}(\mathcal{R}))$ via $< \partial_0, \partial_1 >$. The result follows by applying again Corollary 5.2. □

Two rewriting systems $\mathcal{R} = (X, R)$ and $\mathcal{R}' = (X, R')$ are said to be (k-bounded) equivalent whenever they compute the same (k-bounded) relation on X^*.

PROPOSITION 8.4 *Given a finite rewriting system $\mathcal{R} = (X, R)$, we can decide whether or not it is k-bounded equivalent with a proper subsystem $\mathcal{R}' = (X, R')$, $R' \subseteq R$.*

Proof. It is an immediate consequence of Theorem 7.5. □

9 Pattern Transducers

In this section we study relations obtained by finite mechanisms which associate patterns to sets of patterns. Given two magmoids $M = (M_{m,n})$ and $M' = (M'_{m,n})$, their cartesian product is the magmoid $M \times M'$ defined pointwise

$$(M \times M')_{m,n} = M_{m,n} \times M'_{m,n}$$

for all $m, n \in M$, whereas the $\Diamond$-product ($\Diamond = \circ, \Box$) is given by

$$(a,b)\Diamond(a',b') = (a\Diamond a', b\Diamond b')$$

for all $a \in M_{m,n}$, $a' \in M'_{m,n}$, $b \in M_{n,k}$, $b' \in M'_{n,k}$.

We introduce the magmoid $\mathbb{P}(M)$ (M a magmoid), as follows:

- $\mathbb{P}(M)_{m,n} = \mathcal{P}(M_{m,n})$, the powerset of $M_{m,n}$, $m, n \geq 0$,
- for $Q \subseteq M_{m,n}$, $Q' \subseteq M_{n,k}$, $Q \circ Q' = \{q \circ q' \mid q \in Q, q' \in Q'\}$,
- for $Q \subseteq M_{m,n}$, $Q' \subseteq M_{m',n'}$, $Q \,\Box\, Q' = \{q \,\Box\, q' \mid q \in Q, q' \in Q'\}$.

A *magmoid relation* is a subset $R \subseteq M \times M'$, i.e., $R = (R_{m,n})$ and $R_{m,n} \subseteq M_{m,n} \times M'_{m,n}$, for all $m, n \in \mathbb{N}$. Especially, a magmoid relation of the form $R \subseteq mag(\Sigma) \times mag(\Sigma')$ is called a *pattern relation* from Σ to Σ'. Its associated function

$$\hat{R} : mag(\Sigma) \to \mathbb{P}(mag(\Sigma'))$$

defined by

$$\hat{R}(a) = \{b \mid (a,b) \in R\}$$

is referred to as the *pattern transduction* determined by R. As usual we frequently identify R with $\hat{R}$.

Next we introduce machines computing pattern relations. A pattern transducer is a system

$$\mathcal{T} = (\Sigma, \Sigma', Q, \delta_{\mathcal{T}}, I_{\mathcal{T}}, T_{\mathcal{T}})$$

where

- Σ, Σ' are the doubly ranked input and output alphabets respectively,
- Q is the finite set of states,
- $I_{\mathcal{T}}, T_{\mathcal{T}} \in Rat(Q^*)$, are the initial and final state languages respectively,
- $\delta_{\mathcal{T}}$ is a finite set of moves, that is, triples $(u, \sigma/p_\sigma, v)$, where $u \in Q^m$, $v \in Q^n$, $\sigma \in \Sigma_{m,n}$, $p_\sigma \in mag_{m,n}(\Sigma')$.

The computed relation $|\mathcal{T}| \subseteq mag(\Sigma) \times mag(\Sigma')$ is obtained first by defining the sets $Comp(\mathcal{T})$ and $Loc(\mathcal{T})$ exactly as in the case of pattern automata and then using the canonical projections

$$pr_\Sigma : mag(\delta_{\mathcal{T}}) \to mag(\Sigma), \qquad pr_\Sigma(u, \sigma/p_\sigma, v) = \sigma$$

and

$$pr_{\Sigma'} : mag(\delta_{\mathcal{T}}) \to mag(\Sigma'), \qquad pr_{\Sigma'}(u, \sigma/p_\sigma, v) = p_\sigma.$$

Precisely,

$$|\mathcal{T}| = \{(pr_\Sigma(p), pr_{\Sigma'}(p)) \mid p \in Loc(\mathcal{T})\}.$$

A pattern transducer $\mathcal{T} = (\Sigma, \Sigma', Q, \delta_{\mathcal{T}}, I_{\mathcal{T}}, T_{\mathcal{T}})$ is said to be *k-bounded* when for every $(u, \sigma/p_\sigma, v) \in \delta_{\mathcal{T}}$ it holds

$$|u| \leq k, \ \ |v| \leq k \text{ implies } p_\sigma \text{ is a } k\text{-bounded pattern.}$$

The k-behavior of a k-bounded transducer $\mathcal{T}$, denoted by $|\mathcal{T}|_{(k)}$ is obtained by projecting the pattern language $Loc^{(k)}(\mathcal{T}) = Loc(\mathcal{T}) \cap mag^{(k)}(\delta_{\mathcal{T}})$ via the morphism

$$< pr_\Sigma, pr_{\Sigma'} >: mag(\delta_{\mathcal{T}}) \to mag(\Sigma) \times mag(\Sigma'), < pr_\Sigma, pr_{\Sigma'} > (p) = (pr_\Sigma(p), pr_{\Sigma'}(p)).$$

In other words $|\mathcal{T}|_{(k)} =< pr_\Sigma, pr_{\Sigma'} > (Loc^{(k)}(\mathcal{T}))$.

Remark. For a k-bounded transducer $\mathcal{T}$, clearly we have

$$< pr_\Sigma, pr_{\Sigma'} > (Loc^{(k)}(\mathcal{T})) = |\mathcal{T}| \cap (mag(\Sigma) \times mag(\Sigma'))$$

which is not the case for a general pattern transducer.

Relations of $mag(\Sigma) \times mag(\Sigma')$ (resp. $mag^{(k)}(\Sigma) \times mag^{(k)}(\Sigma')$) obtained in the above way are called recognizable (resp. k-bounded recognizable) pattern relations from Σ to Σ' and the set they constitute is denoted by $Rec(\Sigma, \Sigma')$ (resp. $Rec^{(k)}(\Sigma, \Sigma')$).

The following result is the pattern analog of the well known representation theorem of Nivat (cf. [4]).

PROPOSITION 9.1 *Given a pattern relation $R \subseteq mag(\Sigma) \times mag(\Sigma')$ (resp. $R \subseteq mag^{(k)}(\Sigma) \times mag^{(k)}(\Sigma')$) the following conditions are equivalent*

i) *R is recognizable (resp. k-bounded recognizable);*

ii) *there is a finite doubly ranked alphabet Δ, a strictly alphabetic homomorphism*

$$\varphi : mag(\Delta) \to mag(\Sigma),$$

a homomorphism (resp. k-bounded homomorphism)

$$h : mag(\Delta) \to mag(\Sigma')$$

and a recognizable (resp. k-bounded recognizable) pattern language $L \subseteq mag(\Delta)$ (resp. $L \subseteq mag^{(k)}(\Delta)$) so that

$$R = \{(\varphi(p), h(p)) \mid p \in L\};$$

iii) *R admits a factorization of the form*

$$mag(\Sigma) \xrightarrow{\varphi^{-1}} mag(\Delta) \xrightarrow{-\cap L} mag(\Delta) \xrightarrow{h} mag(\Sigma')$$

(resp.

$$mag^{(k)}(\Sigma) \xrightarrow{\varphi^{-1}} mag^{(k)}(\Delta) \xrightarrow{-\cap L} mag^{(k)}(\Delta) \xrightarrow{h} mag^{(k)}(\Sigma'))$$

where Δ, φ, h and L are as in item ii) above.

Proof. The implication $i) \Rightarrow ii)$ comes immediately from the definition of a pattern transducer.

$ii) \Rightarrow i)$. Assume that Δ, φ, h, L are as in the statement, and consider a pattern automaton $\mathcal{A} = (\Delta, Q, \theta_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$ recognizing L. We construct the pattern transducer

$$\mathcal{T} = (\Sigma, \Sigma', Q, \delta_{\mathcal{T}}, I_{\mathcal{A}}, T_{\mathcal{A}})$$

where $\delta_{\mathcal{T}} = \{(u, \varphi(a)/h(a), v) \mid (u, a, v) \in \theta_{\mathcal{A}}, a \in \Delta\}$.

The pattern homomorphism

$$g : mag(\Delta \times \mathbf{Q}) \to mag(\delta_{\mathcal{T}}), \qquad g(u, a, v) = (u, \varphi(a)/h(a), v)$$

renders commutative the square

$$\begin{array}{ccc} mag(\Delta \times \mathbf{Q}) & \xrightarrow{\ g\ } & mag(\delta_{\mathcal{T}}) \\ {\scriptstyle pr_{\Delta}} \downarrow & & \downarrow {\scriptstyle \langle pr_{\Sigma}, pr_{\Sigma'} \rangle} \\ mag(\Delta) & \xrightarrow[\langle \phi, h \rangle]{} & mag(\Sigma) \times mag(\Sigma') \end{array}$$

where we recall that $\mathbf{Q}$ is the alphabet $\mathbf{Q}_{m,n} = Q^m \times Q^n$, $m, n \geq 0$. From the above diagram we get

$$\begin{aligned} ((\langle \varphi, h \rangle \circ pr_{\Delta})(Loc(\mathcal{A})) &= \langle \varphi, h \rangle (pr_{\Delta}(Loc(\mathcal{A}))) = \\ &= \langle \varphi, h \rangle (L) = \{(\varphi(p), h(p)) \mid p \in L\} \end{aligned}$$

and

$$\begin{aligned} ((\langle pr_{\Sigma}, pr_{\Sigma'} \rangle \circ g(Loc(\mathcal{A})) &= \langle pr_{\Sigma}, pr_{\Sigma'} \rangle (g(Loc(\mathcal{A}))) = \\ &= \langle pr_{\Sigma}, pr_{\Sigma'} \rangle (Loc(\mathcal{T})) = |\mathcal{T}| \end{aligned}$$

and thus $|\mathcal{T}| = \{(\varphi(p), h(p)) \mid p \in L\}$ as wanted.

The equivalence $ii) \Leftrightarrow iii)$ comes from general set theoretic considerations. The k-bounded case is treated in a completely analogous manner.

□

By suitably specifying the data of item ii), of the above characterization result, we get various types of recognizable pattern transductions. More precisely, we say that the relation $R \subseteq mag(\Sigma) \times mag(\Sigma')$ is restricted (resp. non erasing) recognizable whenever there is a finite doubly ranked alphabet Δ, a strictly alphabetic homomorphism $\varphi : mag(\Delta) \to mag(\Sigma)$ and a homomorphism (resp. non erasing homomorphism) $h : mag(\Delta) \to mag(\Sigma')$ so that $R =< \varphi, h > (L)$ for some $L \in RestRec(\Delta)$ (resp. $L \in Rec(\Delta)$). The set of all such relations is denoted $RestRec(\Sigma, \Sigma')$ (resp. $NERec(\Sigma, \Sigma')$).

As we have seen in Section 3, to any pattern automaton $\mathcal{A} = (\Sigma, Q, \theta_{\mathcal{A}}, I_{\mathcal{A}}, T_{\mathcal{A}})$ corresponds the Rec-transduction

$$run_{\mathcal{A}} = \{(pr_{\Sigma}(p), pr_{\mathbf{Q}}(p)) \mid p \in Loc(\mathcal{A})\}$$

(see diagram (c)). Since $pr_{\mathbf{Q}}$ is a strictly alphabetic homomorphism it preserves any recognizable pattern language. Thus, given a recognizable pattern language $L \subseteq mag(\Sigma)$, the pattern language $run_{\mathcal{A}}(L)$ of all runs of patterns in L constitutes a recognizable language as well. An analogous discussion can be made for the k-bounded behavior of k.

PROPOSITION 9.2 *The membership problem is decidable for any pattern automaton in polynomial time.*

Proof.
Let $\mathcal{A}$ be a pattern automaton as above and consider the $Rec^{(k)}$-transduction

$$run_{\mathcal{A}}^{(k)} = \{(pr_{\Sigma}(p), pr_{\mathbf{Q}}(p)) \mid p \in Loc^{(k)}(\mathcal{A})\}.$$

Since for any pattern $p \in mag(\Sigma)$ the singleton $\{p\}$ is k-bounded recognizable for some $k \geq 1$, the pattern language $run_{\mathcal{A}}^{(k)}(\{p\})$ is a k-bounded recognizable language of $mag(\mathbf{Q})$. Clearly, $p \in |\mathcal{A}|$ if and only if $run_{\mathcal{A}}^{(k)}(\{p\}) \neq \emptyset$ and the last fact is polynomially decidable by virtue of Corollary 5.2 of Theorem 5.1.

□

Some examples are cited below.

EXAMPLE 9.3 The diagonal $\Delta \subseteq mag(\Sigma) \times mag(\Sigma)$,

$$\Delta = \{(p,p) \mid p \in mag(\Sigma)\}$$

is clearly a recognizable pattern relation.

EXAMPLE 9.4 The intersection with a recognizable (k-bounded recognizable, restricted recognizable) pattern language as well as arbitrary homomorphisms (k-bounded homomorphisms) are *Rec*- ($Rec^{(k)}$-, RestRec-) transductions. Also inverse strictly alphabetic homomorphisms are *Rec*-, $Rec^{(k)}$- and *RestRec*-tran-sductions.

We say that a class $\mathcal{P}$ of pattern languages is a *Rec*-cone (resp. $Rec^{(k)}$-cone) whenever it is closed under *Rec*-transductions (resp. $Rec^{(k)}$-transductions). Taking into account Proposition 9.1, item iii) a class $\mathcal{P}$ of pattern languages is a *Rec*-cone (resp. a $Rec^{(k)}$-cone) if and only if it is closed under intersection with a recognizable (resp. k-bounded recognizable) pattern language, under arbitrary pattern homomorphisms (resp. k-bounded homomorphisms) and under inverse strictly alphabetic homomorphisms.

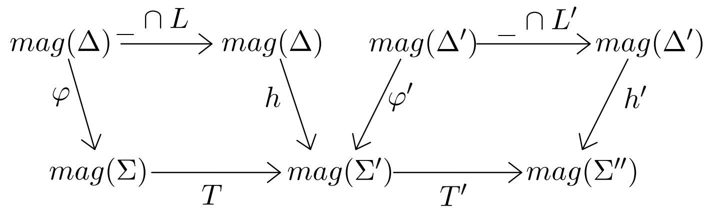
THEOREM 9.5 *The classes of k-bounded recognizable and context free pattern languages constitute $Rec^{(k)}$-cones.*

Proof. The desired results are obtained by virtue of Corollary 4.3 and Theorem 4.5 items iii) and iv) for k-bounded recognizable pattern languages and by Theorem 6.4 and Proposition 6.7 for context free pattern languages. □

An advantage of our notion of pattern transductions is displayed by the following.

THEOREM 9.6 *The composition of two recognizable pattern transductions is again a recognizable pattern transduction.*

Proof. Let $\mathcal{T} : mag(\Sigma) \to mag(\Sigma')$ and $\mathcal{T}' : mag(\Sigma') \to mag(\Sigma'')$ be two *Rec*-transductions, then according to Proposition 9.1 there exist alphabets Δ and Δ', recognizable pattern languages $L \subseteq mag(\Delta)$, $L' \subseteq mag(\Delta')$, strictly alphabetic homomorphisms φ, φ' and homomorphisms h, h', so that $\mathcal{T} = h \circ \_ \cap L \circ \varphi^{-1}$ and $\mathcal{T}' = h' \circ \_ \cap L' \circ \varphi'^{-1}$ respectively. Consider now the following diagram.



We are going to construct an alphabet $\widetilde{\Delta}$, a strictly alphabetic homomorphism

$$\widetilde{\varphi} : mag(\widetilde{\Delta}) \to mag(\Delta),$$

a homomorphism

$$\widetilde{h} : mag(\widetilde{\Delta}) \to mag(\Delta')$$

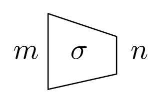
and a recognizable pattern language $\widetilde{L} \subseteq mag(\widetilde{\Delta})$ so that the composition

$$mag(\Delta) \xrightarrow{h} mag(\Sigma') \xrightarrow{\varphi'} mag(\Delta')$$

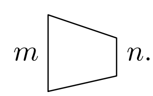
is equal to the composition

$$mag(\Delta) \xrightarrow{\widetilde{\varphi}} mag(\widetilde{\Delta}) \xrightarrow{=\cap \tilde{L}} mag(\widetilde{\Delta}) \xrightarrow{\widetilde{h}} mag(\Delta').$$

The skeleton of an (m, n)-ranked symbol



of the alphabet Σ is the symbol $Sk(\sigma)$:



We denote by $Sk(\Sigma)$ the doubly ranked alphabet of all skeletons of the elements of Σ:

$$Sk(\Sigma) = \{Sk(\sigma) \mid \sigma \in \Sigma\}.$$

Then the canonical pattern homomorphism

$$Sk_\Sigma : mag(\Sigma) \to mag(Sk(\Sigma))$$

sending every $\sigma \in \Sigma$ to its skeleton, associates every pattern $p \in mag(\Sigma)$ to its skeleton $Sk_\Sigma(p)$.

Now, consider the doubly ranked alphabet $\widetilde{\Delta}$ where $\widetilde{\Delta}_{m,n}$ consists of all pairs

$$(\delta, p), \text{ with } \delta \in \Delta_{m,n}, p \in mag_{m,n}(\Delta') \text{ and } Sk_{\Sigma'}(h(\delta)) = Sk_{\Delta'}(p)$$

as well as the strictly alphabetic homomorphism

$$\widetilde{\varphi} : mag(\widetilde{\Delta}) \to mag(\Delta), \qquad (\delta, p) \mapsto \delta$$

and the homomorphism

$$\widetilde{h} : mag(\widetilde{\Delta}) \to mag(\Delta'), \qquad (\delta, p) \mapsto p.$$

We introduce the one state pattern automaton

$$\mathcal{A} = (\widetilde{\Delta}, \{q\}, \theta_{\mathcal{A}}, I_{\mathcal{A}} = \{q\}^*, T_{\mathcal{A}} = \{q\}^*)$$

with $(q^m, (\delta, p), q^n) \in \theta_{\mathcal{A}}$ iff $h(\delta) = \varphi'(p)$.

Then its behavior is just the language $\widetilde{L} \subseteq mag(\widetilde{\Delta})$, consisting of all patterns that can be represented by pairs (π, p), $\pi \in mag(\Delta)$, $p \in mag(\Delta')$ so that $h(\pi) = \varphi'(p)$. The data $\widetilde{\varphi}, \widetilde{h}, \widetilde{L}$ have the desired properties. Now we have

$$\begin{aligned}
\mathcal{T}' \circ \mathcal{T} &= h' \circ _ \cap L' \circ \varphi'^{-1} \circ h \circ _ \cap L \circ \varphi^{-1} \\
&= h' \circ _ \cap L' \circ \widetilde{h} \circ _ \cap \widetilde{L} \circ \widetilde{\varphi}^{-1} \circ _ \cap L \circ \varphi^{-1} \\
&= (h' \circ \widetilde{h}) \circ (_ \cap \widetilde{h}^{-1}(L') \circ (_ \cap \widetilde{L}) \circ (_ \cap \widetilde{\varphi}^{-1}(L)) \circ (\varphi \circ \widetilde{\varphi})^{-1} = \\
&= (h' \circ \widetilde{h}) \circ (_ \cap K) \circ (\varphi \circ \widetilde{\varphi})^{-1}
\end{aligned}$$

where $K = \widetilde{\varphi}^{-1}(L) \cap \widetilde{L} \cap \widetilde{h}^{-1}(L')$.

By Theorem 4.5 i) and Proposition 4.2 the pattern language K is recognizable and thus $\mathcal{T}' \circ \mathcal{T}$ is a *Rec*-transduction and this completes the proof. □

Adopting the above argument in the k-bounded case we obtain the following.

THEOREM 9.7 *The composition of two k-bounded transductions is a k-bounded recognizable transduction.*

The introduced pattern transducers have the disadvantage that they send patterns to *finite* sets of patterns. To eliminate this pathology we consider the following more general machine. A *generalized pattern transducer* is just a system

$$\mathcal{T} = (\Sigma, \Sigma', Q, \delta_{\mathcal{T}}, I_{\mathcal{T}}, F_{\mathcal{T}})$$

where $Q, I_{\mathcal{T}}, F_{\mathcal{T}}$ are as in the definition of a pattern transducer and $\delta_{\mathcal{T}}$ consists of transitions $(u, \alpha/\alpha', v)$, with $u \in Q^m$, $v \in Q^n$, $\alpha' \in mag_{m,n}(\Sigma')$ and $\alpha \in \Sigma_{m,n}$ if $m \neq n$ or else $\alpha \in \Sigma_{n,n} \cup \{\varepsilon_n\}$, where ε_n is the n-th empty word.

In our Nivat-like result (Proposition 9.1) the pattern homomorphism $\varphi : mag(\Delta) \to mag(\Sigma)$ is no longer strictly alphabetic, but it is alphabetic. Our argument in the proof of Theorem 9.6 is still valid for generalized transductions.

THEOREM 9.8 *Generalized pattern transductions are closed under composition.*

It should be noticed that the generalized word transducers (cf. [4, 14]) are instances of the considered model. Indeed let A, A' be ordinary finite alphabets. A *generalized transducer* from A to A' is a 4-tuple $T = (Q, \delta, I, F)$ where $I, F \subseteq Q$ are the initial and final state sets and δ is the finite set of transitions

$$(q, a/w, q'), \qquad q, q' \in Q,\ a \in A \cup \{\varepsilon\},\ w \in A'^*,$$

where ε the empty word. Its behavior $|T|$ is the relation $|T| \subseteq A^* \times A'^*$ formed by all pairs $(a_1 \cdots a_n, w_1 \cdots w_n)$, where there is a successful path

$$q_0 \xrightarrow{a_1/w_1} q_1 \longrightarrow \cdots \longrightarrow q_{n-1} \xrightarrow{a_n/w_n} q_n$$

where $q_0 \in I$, $q_n \in F$ and $a_i \in A \cup \{\varepsilon\}$, $w_i \in A'^*$, $1 \leq i \leq n$. Taking the associated doubly ranked alphabets $\Sigma(A)$ and $\Sigma(A')$ (see Section 3) the above machine is just the generalized pattern transducer

$$\mathcal{T}(T) = (\Sigma(A), \Sigma(A'), Q, \delta, I, F).$$

We should observe that ordinary word transducers can compute pattern relations if we permit the sets I, F to be rational subsets of Q^*. For example take the simple transducer T:

$$q_0 \underset{b/b^2}{\overset{a/a^2}{\rightleftarrows}} q_1.$$

If we choose $I = \{q_0\}$, $F = \{q_1\}$, then its behavior T is the word relation

$$\{(a(ba)^n, a^2(b^2a^2)^n) \mid n \geq 1\}.$$

But if we choose $I = \{q_0^2\}$, $F = \{q_1^2\}$, then $|T|$ is the pattern relation

$$\{\left(\begin{pmatrix} a \\ a \end{pmatrix}\begin{pmatrix} b & a \\ b & a \end{pmatrix}^n, \begin{pmatrix} a^2 \\ a^2 \end{pmatrix}\begin{pmatrix} b^2 & a^2 \\ b^2 & a^2 \end{pmatrix}^n\right) \mid n \geq 1\}.$$

10 Further Research

The present paper can be considered as the basis of developing a theory of cones analogous to the word case. In this perspective the following research directions have to be examined.

1) Determine interesting subcones of the cone CPL of context free pattern languages and detect non-trivial generators for them.

2) Study full AFLs in the present setup.

3) Is it possible to obtain a matrix representation theory of a recognizable pattern transduction?

4) Pattern substitutions can be naturally defined. Given alphabets Σ, Σ' a substitution from Σ to Σ' is just a function $S : \Sigma \to \mathbb{P}(mag(\Sigma'))$. It can be uniquely extended into a morphism of magmoids (denoted by the same letter)

$$S : mag(\Sigma) \to \mathbb{P}(mag(\Sigma')).$$

If $\mathcal{P}$ is a family of pattern languages a $\mathcal{P}$-substitution is a substitution $S : mag(\Sigma) \to \mathbb{P}(mag(\Sigma'))$ such that $\mathcal{S}(\sigma) \in \mathcal{P}$ for all $\sigma \in \Sigma$. For two families of pattern languages $\mathcal{L}$ and $\mathcal{P}$ their composition is the family obtained from the languages of $\mathcal{L}$ by applying $\mathcal{P}$-substitutions. Investigate the above binary operation with respect to cones and full AFLs of pattern languages in analogy with the corresponding operations for families of word languages (cf. [4]).

In an other direction, weighted pattern automata and transducers as well as recognizable pattern series have to be examined.

References

[1] A. Arnold and M. Dauchet, *Théorie des magmoides. I*, RAIRO Theor. Inform. Appl., 12 (1978) 235–257.

[2] A. Arnold and M. Dauchet, *Théorie des magmoides. II*, Theor. Inform. Appl., 13 (1979) 135–154.

[3] D. B. Benson, *The basic algebraic structures in categories of derivations*, Information and Control, 28 (1975) 1–29.

[4] J. Berstel, *Transductions and Context-Free Languages*, Teubner, Stuttgart, 1979.

[5] F. Bossut, M. Dauchet and B.Warin, *A Kleene theorem for a class of planar acyclic graphs*, Inform. and Comput., 117 (1995) 251–265.

[6] S. L. Bloom and Z. Ésic, *Iteration Theories*, EATCS Monographs on Theoretical Computer Science, Springer-Verlag, 1991.

[7] S. Bozapalidis and A. Kalampakas, *An Axiomatization of Graphs*, Acta Inform., 41 (2004) 19–61.

[8] A. Corradini and F. Gadducci, *An algebraic presentation of term graphs, via gs-monoidal categories*, Appl. Categ. Structures, 7 (1999) 299–331.

[9] V. Claus, *Ein Vollständigkeitssatz für Programme und Schaltkreise*, Acta Inform., 1 (1971) 64–78.

[10] S. Eilenberg and J. B. Wright, *Automata in General Algebras*, Information and Control, 11 (1967) 452–470.

[11] J. Engelfriet, *Context-free graph grammars*, in: G. Rozenberg and A. Salomaa (Eds.), Handbook of Formal Languages. Vol. III: Beyond Words, Chapter 3, 125–213, Springer, 1997.

[12] J. Engelfriet, *Graph grammars and tree transducers*, in Proceedings of CAAPŠ94 (S.Tison, ed.), Lecture Notes in Comput. Sci., 787 (1994) 15–36.

[13] J. Engelfriet and J. J. Vereijken, *Context-free graph grammars and concatenation of graphs*, Acta Inform., 34 (1997) 773–803.

[14] S. Ginsburg, *The mathematical theory of context-free languages*, McGraw-Hill Book Co., 1966.

[15] F. Gécseg and M. Steinby, *Tree Automata*, Akademiai Kiado, Budapest, 1984.

[16] F. Gécseg and M. Steinby, *Tree Languages*, Handbook of Formal Languages, vol. 3, 1–68 (1997).

[17] J. Gibbons, *An initial-algebra approach to directed acyclic graphs*, in Mathematics of program construction (Kloster Irsee, 1995), Lecture Notes in Comput. Sci., 947 (1995) 282–303.

[18] A. Habel, *Hyperedge Replacement: Grammars and Languages*, Lecture Notes in Comput. Sci., 643 (1992) 143–158.

[19] A. Habel and H.-J. Kreowski, *May we introduce to you: hyperedge replacement*, in Graph-Grammars and their Application to Computer Science (H.Ehrig, M.Nagl, G.Rozenberg, A.Rosenfeld, Eds.), Lecture Notes in Comput. Sci., 291 (1987) 15–26.

[20] G. Hotz, *Eine Algebraisierung des Syntheseproblems von Schaltkreisen*, EIK, 1 (1965) 185–205, 209-231.

[21] G. Hotz, *Eindeutigkeit und Mehrdeutigkeit formaler Sprachen*, EIK, 2 (1966) 235–246.

[22] T. Kamimura and G. Slutzki, *Parallel and two-way automata on directed ordered acyclic graphs*, Information and Control, 49 (1981) 10–51.

[23] T. Kamimura and G. Slutzki, *Transductions of DAGS and trees*, Math. Syst. Theory, 15 (1982) 225–249.

[24] C. P. Schnorr, *Transformational Classes of Grammars*, Information and Control 14 (1969) 252–277.