On the transition graphs of automata and grammars

Didier Caucal, Roland Monfort

UNITÉ DE RECHERCHE
INRIA-RENNES

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P. 105
78153 Le Chesnay Cedex
France
Tél.:(1) 39 63 55 11

Rapports de Recherche

N° 1318

Programme 1
Programmation, Calcul Symbolique et Intelligence Artificielle

ON THE TRANSITION GRAPHS OF AUTOMATA AND GRAMMARS

Didier CAUCAL
Roland MONFORT

Octobre 1990

RR-1318

IRISA INSTITUT DE RECHERCHE EN INFORMATIQUE ET SYSTEMES ALEATOIRES

Campus Universitaire de Beaulieu
35042 - RENNES CEDEX
FRANCE
Téléphone : 99.36.20.00
Télex : UNIRISA 950 473F
Télécopie : 99.38.38.32

On the transition graphs of automata and grammars

Didier CAUCAL and Roland MONFORT

Publication interne n° 551 - Septembre 1990 - 46 pages

Abstract . The transition graph of a pushdown automaton (resp. a context-free grammar) is the set of transitions accessible from its initial configuration (resp. from its axiom). Every grammar transition graph is a pushdown transition graph, but the converse is false. Nevertheless, we can decide if a pushdown transition graph is a grammar transition graph, and we can then transform the automaton into a grammar with the same transition graph.

Sur les graphes de transition des automates et des grammaires

Résumé . Le graphe de transition d'un automate à pile (resp. d'une grammaire algébrique) est l'ensemble des transitions accessibles à partir de sa configuration initiale (resp. de son axiome). Tout graphe de transition d'une grammaire est le graphe de transition d'un automate, mais la réciproque est fausse. Cependant, on sait décider si le graphe de transition d'un automate est le graphe de transition d'une grammaire, et on sait alors transformer l'automate à pile en une grammaire algébrique de même graphe de transition.

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE
(U.R.A. 227)
UNIVERSITE DE RENNES I I.N.S.A. DE RENNES

INSTITUT NATIONAL DE RECHERCHE
EN INFORMATIQUE ET EN AUTOMATIQUE
(UNITE DE RECHERCHE DE RENNES)

On the transition graphs of automata and grammars (1)

Didier CAUCAL and Roland MONFORT
IRISA and IRMAR , Campus de Beaulieu
F35042 Rennes-Cedex , France

Introduction

A pushdown automaton on the sets Σ of terminals, X of stack letters and Q of states, is a finite set of rules $pA \xrightarrow{f} q\alpha$ labelled by $f \in \Sigma \cup \{\varepsilon\}$, making the automaton move from state p to state q , and the stack letter A change into the stack word α . Such a rule defines the automaton transitions $pA\beta \xrightarrow{f} q\alpha\beta$, replacing the top A of the stack $A\beta$ by the word α . The set of transitions obtained from any axiom in $Q.X^*$ is a pushdown transition graph or a context-free graph in the sense of [Mu-Sc 85] .

A context-free grammar may be viewed as a pushdown automaton with one state, which can be omitted because it is irrelevant. The grammar rules have then the form $A \xrightarrow{f} \alpha$, and any transition graph is called alphabetic.

Therefore, every alphabetic graph is a pushdown transition graph. But there exist pushdown transition graphs which are not alphabetic.

First, we show (in section 1) that pushdown transition graphs can effectively be generated from deterministic graph grammars. From this, we can decide (in section 2) if a pushdown transition graph is alphabetic, and we can then transform this automaton into a context-free grammar with the same transition graph.

Last, we show (in section 3) that there exist pushdown transition graphs whose structure is

(1) This work was presented at WG 90 and will appear in LNCS .

comparable with no alphabetic graph, in the bisimulation meaning [Pa 81] . To do so, we consider their canonical graphs obtained as quotient by their greatest self-bisimulation. Then and in an effective way, we establish that the canonical graphs of the alphabetic graphs are always alphabetic. But, we give a pushdown transition graph whose canonical graph is not a pushdown transition graph.

Section 1 has been detailed in [Ca 90 b] . The proofs in section 2 are given in the appendix. Section 3 is developped in [Ca 90 a] .

1. Pushdown transition graphs

In this section, we define the pushdown transition graphs and state [Ca 90 b] that such graphs have a regular structure. They are the rooted graphs (a root is a vertex from which one can access to all the other vertices of the graph) of finite degree (i.e. every vertex belongs only to a finite number of arcs) obtained by iteration from a finite family of finite graphs, and hence called pattern graphs. Furthermore, this characterization is effective. One can transform the transitions of any automaton into a system of patterns generating its transition graph. Reciprocally, every system of patterns generating a rooted graph G of finite degree, can be transformed into an automaton whose graph is isomorphic to G .

Prefix transition graph

A pushdown automaton (pda) is a special case of word rewriting system. In the sequel, X is an alphabet of *non-terminals* and L is a set of *labels*. A (labelled) rewriting system R on (X,L) is a finite set of rules $u \xrightarrow{f} v$ where $f \in L$ and $u , v \in X^*$.

Definition. A *pushdown automaton* (without initial and final states but with extended reading) is a rewriting system R on (X,L) , satisfying the following conditions :

(i) X is partitioned into $Q_R \cup P_R$

(ii) for any rule $u \xrightarrow{f} v$ in R , we have $u \in Q_R.P_R$ and $v \in Q_R.P_R{}^*$.

The rewritings of a system R are defined as applications of the rewriting rules in any context : a *step of rewriting* [resp. a *step of prefix rewriting*] is a labelled transition $xuy \xrightarrow{f} xvy$ [resp. $uy \overset{f}{\longmapsto} vy$] where $u \xrightarrow{f} v$ is a rule in R and $x , y \in X^*$.

The *accessible transition graph* T(R,r) [resp. the *accessible prefix transition graph* P(R,r)] generated by R from an axiom $r \in X^*$ is the set of arcs $w \xrightarrow{f} z$ [resp. $w \overset{f}{\longmapsto} z$] such that w is obtained by rewriting [resp. prefix rewriting] from r .

Moreover, if R is an automaton and $r \in Q_R.P_R^*$ then T(R,r) = P(R,r) is the transition graph of the automaton R accessible from r : any graph isomorphic to a such one is called a *pushdown transition graph,* where the isomorphism is a renaming of the vertices. Figure A gives an example of a pushdown transition graph.

Let R be the automaton on ({A,B,p,q} , {a,b,c,d}) defined as follows :

$$R = \{ pA \xrightarrow{a} qA \ , \ pA \xrightarrow{b} pBA \ , \ pB \xrightarrow{c} p \ , \ pB \xrightarrow{a} qB \ , \ pB \xrightarrow{b} pBB \ , \ qB \xrightarrow{d} q \ \} .$$

The accessible transition graph T(R,pA) = P(R,pA) is represented by

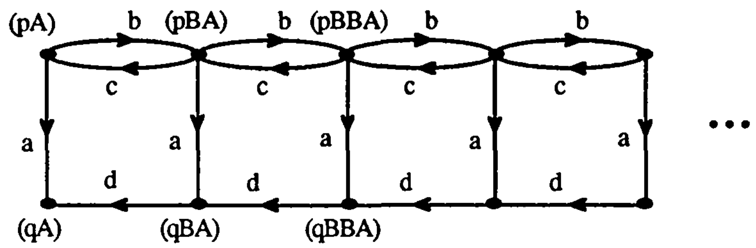


Fig. A . Pushdown transition graph .

Hence every pushdown transition graph is an accessible prefix transition one. The converse holds too [Ca 90 b] .

Proposition 1.1 . *Prefix transition graphs coincide with pushdown transition graphs.*

Deterministic graph grammar

Let us notice that every accessible prefix transition graph P(R,r) has root r and is of finite degree. Muller and Schupp [Mu-Sc 85] have shown that pushdown transition graphs have a regular structure: they are the rooted graphs of finite degree which have a finite number of non-isomorphic connected components obtained after removing all vertices within a distance d of a given vertex, for any d . Thus, pushdown transition graphs may be cut into slices of a finite number of 'patterns'. We relax the constraint of splitting up the graph 'by slices' and allow to remove patterns of arbitrary shapes and sizes. This adds nothing to Muller and Schupp decomposition [Mu-Sc 85] , but it is easier to construct patterns for pushdown transition graph.

To begin with, let us introduce patterns and their gluing. In order to ease the presentation, we use graph grammars, and first recall their definition.

Definition. A *graph grammar* on a graded alphabet $F = \cup_n F_n$ and set of vertices V , is a finite set of hyperarc replacement rules $fv_1 \ldots v_n \rightarrow H$ where the word $fv_1 \ldots v_n$ is a *hyperarc* labelled by the *non-terminal* $f \in F_n$, the v_i are distinct vertices and H is a finite *hypergraph*, that is a set of hyperarcs. Every *terminal* of the grammar, that is to say every label of a right member rule hyperarc which is not a non-terminal, is of arity 2 .

A graph grammar is *deterministic* if there is only one rule for each non-terminal f .

Figure B is an example of a deterministic graph grammar.

Let A , a , b be in F of respective arity 3 , 2 , 2 .

Let G = { (A123 , {a12,a14,a25,b63,A564}) } be a deterministic graph grammar.

A is the unique non-terminal of G , and G is represented as follows :

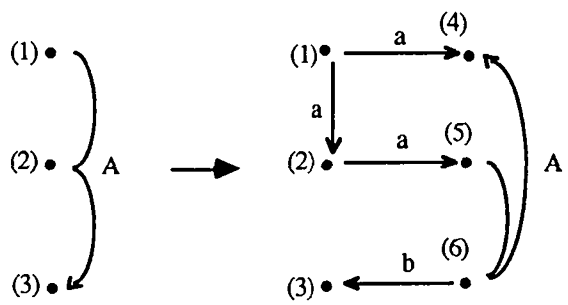


Fig. B . Deterministic graph grammar .

Each deterministic graph grammar defines a graph, resulting from a given start graph by iterating the graph rewriting [Ha-Kr 87] . Intuitively, a rewriting step consists in choosing a hyperarc $ft_1 \ldots t_n$ whose label f indicates the rule $fs_1 \ldots s_n \rightarrow H$ to use, and the vertices s_i in H indicate how to replace $ft_1 \ldots t_n$ by a copy of H . We use the symbol + for the disjoint union of sets.

Definition. Given a graph grammar G on (F,V) and a hypergraph M on (F,V) , M gives a hypergraph N in one *rewriting step*, and we note $M \rightarrow_G N$, if for some rule $fs_1 \ldots s_n \rightarrow H$, there exists a hypergraph M' such that $M = M' + \{ft_1 \ldots t_n\}$ and $N = M' + \{ hg(x_1) \ldots g(x_m) \mid hx_1 \ldots x_m \in H \}$ for some matching function g mapping s_i to t_i , and mapping injectively the other vertices of H to vertices outside of M .

Note that $\rightarrow_G$ is not in general a functional relation, even though G is deterministic. Nevertheless, if we let $M \rightarrow_{G,X} N$ denote the rewriting of a non-terminal hyperarc X , then

$$M \rightarrow_{G,X_1} \circ \ldots \circ \rightarrow_{G,X_n} N \text{ if and only if } M \rightarrow_{G,X_{\pi(1)}} \circ \ldots \circ \rightarrow_{G,X_{\pi(n)}} N$$

for any $X_i \in M$, and for any permutation π on $\{1,\ldots,n\}$. Thus, it makes sense to define steps of complete parallel rewriting $M \Rightarrow_G N$ as follows :

$$M \Rightarrow_G N \text{ if } M \rightarrow_{G,X_1} \circ \ldots \circ \rightarrow_{G,X_n} N ,$$

and M has exactly the n non-terminal hyperarcs $X_1 , \ldots , X_n$. One step of complete parallel rewriting corresponds to the Kleene substitution. On that basis, we define below $G^{\omega}(M)$, the set of hypergraphs generated from the axiom M according to the deterministic graph grammar G , where $[M] = \{ fst \in M \mid f \text{ is a terminal} \}$ is the set of terminal arcs of M .

Definition. $G^{\omega}(M)$ is the set of hypergraphs $N = \cup_n [N_n]$ where $(N_n)_{n \geq 0}$ is an infinite sequence of hypergraphs such that $N_0 = M$ and $N_n \Rightarrow_G N_{n+1}$ for all n .

Since G is deterministic, $G^{\omega}(M)$ has a single element up to hypergraph isomorphism. When M is finite, this element is called the *pattern graph* generated by G from M . Pattern graphs are the equational graphs of Bauderon and Courcelle [Ba 89] , [Co 89 a] , [Co 89 b] . The grammar of the figure B generates from A123 the pattern graph of the following figure C.

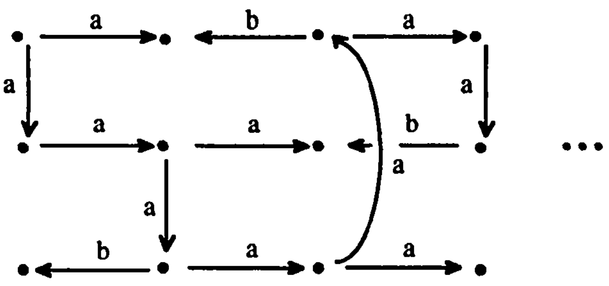


Fig. C . Pattern graph .

Let us point out that every finite degree pattern graph is a bounded degree graph, i.e. there exists a bound b such that for every vertex s , the number of arcs to which s belongs is smaller than b .

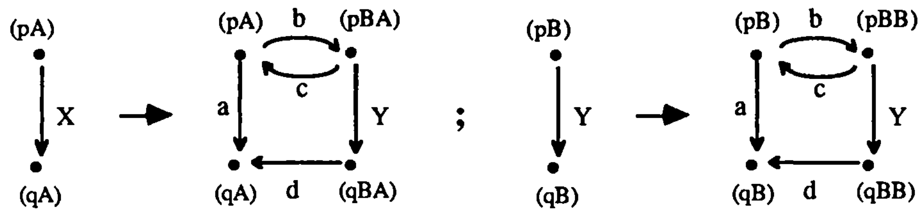
Pushdown transition graphs as pattern graphs

The structure of a pushdown transition graph can be described in an effective way [Ca 90 b] by patterns.

Theorem 1.2 . *Any pair* (R,r) *of a word rewriting system* R *and a non-terminal word* r , *may be effectively transformed into a pair* (G,M) *of a deterministic graph grammar* G *and a hyperarc* M , *such that the corresponding graphs* P(R,r) *and* $G^{\omega}(M)$ *are isomorphic.*

The construction of the grammar G from (R,r) is made by generating P(R,r) by vertices of growing length, and connected components. From figure A , we obtain the deterministic graph grammar of the following figure D .

The following deterministic graph grammar



generates from hyperarc X the accessible prefix transition graph of figure A .

Fig. D . Extraction of a graph grammar from an accessible prefix transition graph .

The converse of theorem 1.2 remains true [Ca 90 b] if we restrict ourselves to rooted graphs of finite degree.

Theorem 1.3 . *Any pair* (G,M) *of a deterministic graph grammar* G *and an axiom* $M = fs_1...s_n$, *such that* $G^\omega(M)$ *has finite degree and root* s_1 , *may be effectively transformed into a pair* (R,r) , *of a word rewriting system* R *and an axiom* r , *such that the corresponding graphs* $G^\omega(M)$ *and* P(R,r) *are isomorphic.*

First, we put G in Greibach form, that is to say for every rule $gt_1...t_n \to H$ of G , all vertices of all non-terminal hyperarcs of H are distinct and are different of $t_1,...,t_n$. Then, every terminal arc of H gives a rewriting rule of R such that each vertex of the prefix transition graph of R is of the form su where s is the place of the vertex in its pattern and u is the mirror path of non-terminals needed to obtain the vertex. Applied to the grammar of figure D , this construction gives the following rewriting system R :

$$R = \{ [pA]X \xrightarrow{a} [qA]X \ , \ [pA]X \xrightarrow{b} [pB]YX \ , \ [pB]YX \xrightarrow{c} [pA]X \ , \ [qB]YX \xrightarrow{d} [qA]X \ ,$$

$$[pB]Y \xrightarrow{a} [qB]Y \ , \ [pB]Y \xrightarrow{b} [pB]YY \ , \ [pB]YY \xrightarrow{c} [pB]Y \ , \ [qB]YY \xrightarrow{d} [qB]Y \ \} ,$$

and P(R,[pA]X) is isomorphic to the graph of figure A . A new proof of theorem 1.3 is given in the appendix.

From the two former theorems and from proposition 1.1 follows a characterization of the pushdown transition graphs as pattern graphs.

Corollary 1.4 . *Pushdown transition graphs coincide exactly with rooted pattern graphs of finite degree.*

Let us point out that the two former theorems allow the transformation of an automaton into another one, by considering graph grammars of their transition graphs. Such a transformation is carried out in the following section to characterize the alphabetic graphs defined hereafter.

2. Alphabetic graphs

In this section, we define the left derivation graphs of the context-free grammars in Greibach form. We show they form a proper subclass of the pushdown transition graphs, and we give an effective characterization of it. One can decide if a corooted pushdown transition graph (a coroot is a vertex accessible from any vertex of the graph) is a left derivation graph of a context-free grammar with the same prefix transition graph. This result is then extended to pushdown transition graphs which are finitely coaccessible, i.e. with a finite set of vertices accessible from any other vertex. The proofs of this section are given in the appendix.

A proper subclass of the pushdown transition graphs

Let us consider the context-free grammars where the right members of the rules are in Σ^*X^* (Σ : terminals , X : non-terminals) , and their associated left derivation graphs.

Definition. An *alphabetic system* is a rewriting system R such that for each rule $u \xrightarrow{f} v$ in R , u is a letter.

An *alphabetic graph* is the prefix transition graph P(R,r) of an alphabetic system R accessible from any axiom r . From proposition 1.1 , any alphabetic graph P(R,r) is isomorphic to a pushdown transition graph P(S,s) : anyway, it suffices to take a new symbol p and to define

$$S = \{ pA \xrightarrow{f} pv \mid A \xrightarrow{f} v \in R \} \text{ and } s = pr .$$

Let us show that the converse is false : there exists a pushdown transition graph that is not alphabetic. To do so, we need some notations and definitions.

A *path* u in a graph is a non-empty word on the vertices such that there exists an arc of source u(i) and target u(i+1) for $1 \leq i < |u|$, where $|u|$ is the length of u . A path u has the source u(1) and the target u(|u|) . We say a vertex s is a *multiple start* for the vertices t and t' if there are two paths u and v of source s , and respective targets t and t' , without common vertex other than the first one, and also the last one if $t = t'$, i.e.

$$u(i) = v(j) \Rightarrow (i = j = 1) \vee (i = |u| \wedge j = |v|).$$

Beware of the fact that any vertex is a multiple start of itself and of its successors, i.e. if there is an arc of source s and target t then s is a multiple start of t .

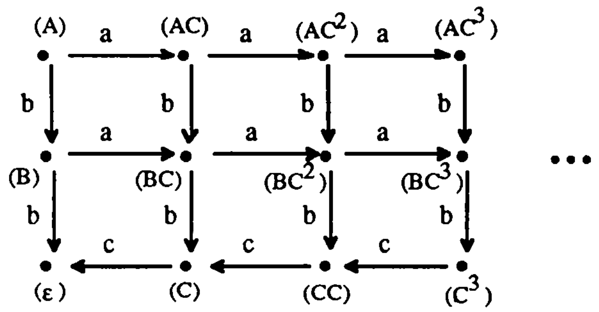
A graph is of *finite multiplicity* (resp. of *bounded multiplicity* for an integer b) if any vertices t and t' have only a finite number (resp. less than b) multiple starts.

Figure E gives an example of an alphabetic graph of finite but not bounded multiplicity.

Let R be the alphabetic system on ({A,B,C},{a,b,c}) defined as follows:

$$R = \{ A \xrightarrow{a} AC , A \xrightarrow{b} B , B \xrightarrow{a} BC , B \xrightarrow{b} \varepsilon , C \xrightarrow{c} \varepsilon \} .$$

The alphabetic graph P(R,A) is represented by



is of finite multiplicity , but not bounded because $A , \ldots , AC^n , BC^{n-1} , BC^n$ are multiple starts of BC^n .

Fig. E . Alphabetic graph of not bounded multiplicity .

On the contrary, the pushdown transition graph in figure A is not of finite multiplicity : pB^+A is a set of multiple starts for qA . Such a graph is not alphabetic, as states the following proposition.

Proposition 2.1 . *Every alphabetic graph is of finite multiplicity.*

Corollary 2.2 . *The set of alphabetic graphs is a proper subclass of the set of pushdown transition graphs.*

A characterization of the alphabetic graphs

We characterize the alphabetic graphs as pushdown transition graphs and we show that the condition of finite multiplicity is sufficient.

Conjecture 2.3 . *Every pushdown transition graph of finite multiplicity is an alphabetic graph.*

Henceforth $G = P(R,r)$ is the pushdown transition graph of R from the axiom r . From theorem 1.2, one can transform (R,r) into a deterministic graph grammar H generating $P(R,r)$ from a hyperarc M. We want to solve the conjecture by showing that in case G is of finite multiplicity, one can transform the grammar H into a so-called grammar with constrained returns, defined below.

Definition. A deterministic graph grammar is *with constrained returns* if for any rule $fv_1 \ldots v_n \rightarrow K$ and for every non-terminal hyperarc $gx_1 \ldots x_m$ in K , if x_i and x_j are sources of terminal arcs in K then $i = j$.

The interest lies in the fact that every pushdown transition graph generated by a grammar with constrained returns is an alphabetic graph.

Proposition 2.4 . *Any pair* (H,M) *of a grammar* H *with constrained returns and axiom* $M = fs_1 \ldots s_n$ *such that* $H^{\omega}(M)$ *has finite degree and root* s_1 , *may effectively be transformed into a pair* (R,r) *of an alphabetic system* R *with axiom* r , *such that the graphs* $H^{\omega}(M)$ *and* $P(R,r)$ *are isomorphic.*

First, we name the vertices in $H^{\omega}(M)$ as in theorem 1.3 . Then, we remove the names of the vertices of constrained returns and we rewrite every path $X_1 \ldots X_n$ as $(X_1,X_2) \ldots (X_{n-1},X_n)X$ where X is a new symbol.

To each transition $A\beta \xrightarrow{a} \alpha\beta$ of $H^{\omega}(M)$ corresponds an alphabetic rule $A \xrightarrow{a} \alpha$. Applied to the grammar of figure B , this construction gives the following alphabetic system R :

$$R = \{ X \xrightarrow{a} 1(A,A)X \ , \ 1 \xrightarrow{a} 3(A,A) \ , \ 1 \xrightarrow{a} \varepsilon \ , \ (A,A) \xrightarrow{a} 1(A,A)^2 \ , \ (A,A) \xrightarrow{b} 3 \ \} ,$$

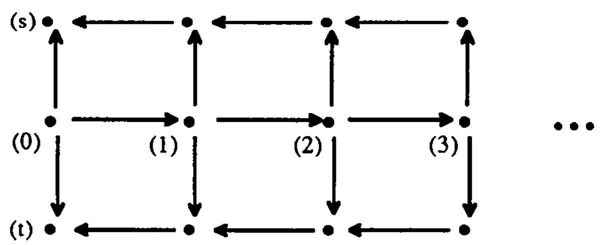
and P(R,1X) is isomorphic to the graph of figure C .

To get a grammar with constrained returns generating G = P(R,r) , we begin by restricting ourselves to the case where G has a coroot c . For such a graph, the notion of finite multiplicity is simpler : we always take t = t' in the definition.

Proposition 2.5 . *A corooted graph has a finite multiplicity if and only if every vertex has only a finite number of multiple starts.*

Figure F shows that proposition 2.5 is false if the graph has no coroot.

Let us consider the following unlabelled (with one label only) pattern graph :



The vertex 0 is a root of this graph of finite degree. From corollary 1.4 , this graph is a pushdown transition graph. Every vertex has at most three multiple starts. But {0,1,2,3,...} is an infinite set of multiple starts for {s,t} . From proposition 2.1 , this graph is not alphabetic.

Fig. F . A non alphabetic graph .

Moreover, if G is an alphabetic graph then every path u from $\beta\alpha$ to c , where α has a length greater than or equal to the length of c, goes through α, i.e. there exists $1 \leq i \leq |u|$ such that $u(i) = \alpha$; we say that the vertex $\beta\alpha$ is codominated by the vertex α.

Definition. In a graph G with coroot c, a vertex s *codominates* a vertex t, or s is a *codominator* of t, and we note $s \leq t$, if every path from t to c goes through s.

When G is of finite multiplicity, this *codomination* relation allows the construction of a so-called codomination grammar, generating G.

Definition. A *codomination grammar* H is a deterministic graph grammar for which there exists an infinite sequence of hypergraphs $(N_n)_{n\geq 0}$ where $N_n \Rightarrow_G N_{n+1}$ and $G = \bigcup_n[N_n] = H^\omega(N_0)$, such that every connected component C of $G - N_n$ has all its vertices codominated by one vertex, common to N_n and to C.

Figure G gives an example of a codomination grammar.

Hence, every codomination grammar has constrained returns, and by virtue of proposition 2.4, generates alphabetic graphs.

Previously to the construction of a codomination grammar, we give some basic properties of this relation.

We denote by $<$ and $\prec$ the *strict and direct codomination relations*, i.e.

$$s < t \text{ if } s \leq t \wedge s \neq t$$

and $$s \prec t \text{ if } s < t \wedge \forall r\,(s < r \leq t \Rightarrow r = t).$$

Let $\rightarrow$ be the *successor relation* on the vertices of the graph, i.e. $s \rightarrow t$ if there is an arc of source s and target t. The *valuation* $v(s)$ of a vertex s of G is the shortest length minus one (the number of arcs) of the paths of source s and target c, i.e. $v(c) = 0$ and $v(s) = \min\{\, v(t) + 1 \mid s \rightarrow t \,\}$ for $s \neq c$.

The codomination relation orders the vertices of the graph in a tree with growing valuations.

Proposition 2.6 . *Let us consider a graph with coroot* c *. The codomination relation* $\leq$ *is an order relation with least element* c *, and strictly growing valuations, i.e.*

$$s < t \;\Rightarrow\; v(s) < v(t)\,.$$

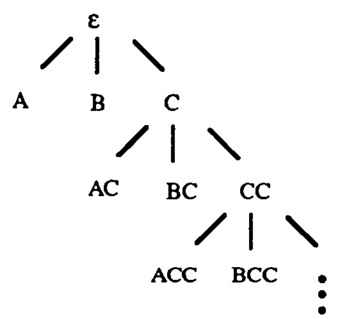
The direct codomination relation $-\!\!\!\prec$ *defines a tree structure on the graph vertices, i.e.*

$$(\, s \leq r \;\wedge\; t \leq r \,) \;\Rightarrow\; (\, s \leq t \;\vee\; t \leq s \,)\,.$$

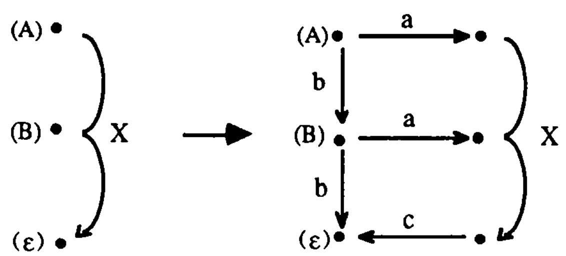
This tree, called codomination tree, is of finite degree if and only if the graph has finite multiplicity.

Figure G gives an example of codomination tree where : father $-\!\!\!\prec$ son .

Let G be the graph in figure E , with coroot ε . Its codomination tree is the following one :



The graph grammar shown hereafter



is a codomination grammar for G .

Fig. G . Codomination tree and grammar .

We infer the last property of proposition 2.6 from the first equivalence of the lemma hereafter. The set of multiple starts of s is denoted by $Dm\{s\}$. Let $E_t = \{ r \mid v(r) < v(t) \wedge t \in Dm\{r\} \}$ be the set of vertices with t as multiple start and valuation less than $v(t)$.

Lemma 2.7 . *The direct codominant of* $t \neq c$ *is the strict codominant of* t *with* t *as multiple start, i.e.* $s \prec\!\!\!-\, t$ *iff* $s < t \wedge t \in Dm\{s\}$.

It is also the vertex of minimum valuation with t *as multiple start, i.e.*

$$s \prec\!\!\!-\, t \text{ iff } s \in E_t \wedge \forall r \in E_t ,\ v(s) \leq v(r) .$$

From a graph grammar generating G (cf. theorem 1.2), we can decide if a vertex t is a multiple start of a vertex s , and from the second equivalence of the lemma 2.7 , we can decide $s \prec\!\!\!-\, t$ and $s \leq t$.

Proposition 2.8 . *Given a triple* (R,r,c) *of an automaton* R *, an axiom* r *and a coroot* c *of the accessible transition graph* $P(R,r)$ *, the relations of codomination and strict codomination are decidable.*

The difficulty lies in deciding that t is a multiple start of s . From theorem 1.2 , we construct a deterministic graph grammar G and a hyperarc M such that $P(R,r)$ is the pattern graph $G^{\omega}(M)$. We establish the existence of a bound b , depending on s and t , such that t is a multiple start of s in the finite graph $G^b(M)$ obtained from M by b parallel rewriting steps, if and only if t is a multiple start of s in $G^{\omega}(M)$.

Proposition 2.8 allows the construction of a codomination grammar, when G has finite multiplicity. In such a graph, we have a partition of the arcs of source r in two classes : the arcs whose target is codominated by r and the others. To every vertex r , one associates the pattern M_r containing the arcs of source r and target codominated by r , and the arcs of source s directly codominated by r and target not codominated by s , i.e.

$$M_r = \{ r \xrightarrow{f} t \in G \mid r \leq t \} \cup \{ s \xrightarrow{f} t \in G \mid r \prec\!\!\!-\, s \wedge \neg(s \leq t) \}.$$

So M_s contains in particular all the arcs of source s which are not in M_r . The process will then reach every arc in G . Then for all $n \geq 0$, one constructs the subgraph G_n of G , union of the patterns

M_r where c codominates r by at most n-1 steps of direct codomination, i.e.

$$G_n = \{ s \xrightarrow{f} t \in G \mid \exists i,\ 0 \leq i \leq n\text{-}1,\ c \prec^i s \} \cup \{ s \xrightarrow{f} t \in G \mid c \prec^n s \wedge \neg(s \leq t) \}.$$

We easily verify that

$$G - G_n = \{ s \xrightarrow{f} t \in G \mid \exists r,\ c \prec^n r \wedge r \leq s \wedge r \leq t \} \text{ for all } n \geq 0,$$

and that every connected component of $G - G_n$ (dotted lines in fig. H) is the restriction of G to the vertices codominated by the same vertex r such that $c \prec^n r$. In figure H, we give a scheme of this construction.

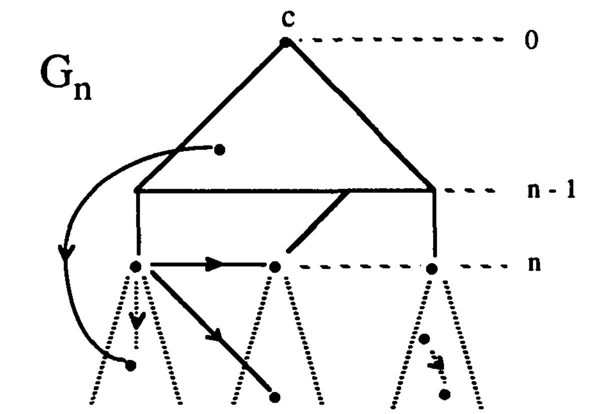


Fig. H . Graph ordered by codomination .

The set of connected components of $G - G_n$ is finite, and by transforming the graph grammar obtained from theorem 1.2 , we have a codomination grammar for G .

> **Proposition 2.9 .** *Given a triple* (R,r,c) *of an automaton* R , *an axiom* r *and a coroot* c *of the transition graph* P(R,r) , *we can decide if* P(R,r) *is of finite multiplicity, in which case, we construct a codomination grammar of* P(R,r) .

Propositions 2.4 and 2.9 establish conjecture 2.3 restricted to corooted graphs, in an effective way.

Proposition 2.10 . *Given a triple* (R,r,c) *of an automaton* R , *an axiom* r *and a coroot* c *of the transition graph* P(R,r) , *we can decide if* P(R,r) *is alphabetic, in which case, we can transform* (R,r,c) *into a pair* (S,s) *where* S *is an alphabetic system and* P(S,s) *is isomorphic to* P(R,r) .

Now, we can extend proposition 2.10 to any *finitely coaccessible* graph, i.e. such that there exists a finite set A of vertices which is accessible from any vertex s , that is $\exists\, t \in A$, $s \rightarrow^* t$.
If G is a graph finitely coaccessible from $\{c_1,\ldots,c_n\}$ then we complete G in a graph

$$G' = G \cup \{\ c_i \xrightarrow{f} c \mid 1 \leq i \leq n\ \}$$

where c is not a vertex of G . Then c is a coroot of G' and G' is of finite multiplicity if and only if G is of finite multiplicity. Proposition 2.10 applied to G' is then extended to every finitely coaccessible graph.

Theorem 2.11 . *Given a triple* (R,r,A) *of an automaton* R , *an axiom* r *and a finite set* A *of vertex, such that the transition graph* P(R,r) *is finitely coaccessible from* A , *we can decide if* P(R,r) *is alphabetic, in which case, we can transform* (R,r,A) *into a pair* (S,s) *where* S *is an alphabetic system and* P(S,s) *is isomorphic to* P(R,r) .

Moreover, given a rewriting system R and an axiom r , from theorem 1.2 and the monadic second order logic on pattern graphs [Mu-Sc 85] and [Co 89 a] , we can decide if P(R,r) has a coroot or is finitely coaccessible, but in this case, we do not know if there exists an algorithm to produce such a finite coaccessible set.
Conjecture 2.3 is then established in an effective way, for every finitely coaccessible graph. Yet, it remains open for pushdown transition graph not finitely coaccessible.

3. Canonical graphs

In the former section, we have shown that the alphabetic graphs have a simpler structure than the pushdown transition graphs. It follows that properties not satisfied for the class of pushdown transition graphs become true when we limit ourselves to the class of alphabetic graphs. The class of alphabetic graphs with terminal coroot (without outgoing arc) is closed for the quotient by the greatest self-bisimulation [Ca 90 a] . On the contrary, there exists a pushdown transition graph whose quotient by the greatest self-bisimulation is not a pushdown transition graph (cf. figure K).

Let us recall the notions of graph bisimulation and canonical graph. Graph bisimulation defined below, was introduced by Park [Pa 81].

Definition. A *bisimulation* $G \leftrightarrow H$ is a relation R on vertices, which is total and onto, and satisfies the following two conditions :

(i) $s\,R\,t \wedge (s \xrightarrow{f} s') \in G \;\Rightarrow\; \exists\, (t \xrightarrow{f} t') \in H,\; s'\,R\,t'$

(ii) $s\,R\,t \wedge (t \xrightarrow{f} t') \in H \;\Rightarrow\; \exists\, (s \xrightarrow{f} s') \in G,\; s'\,R\,t'$.

Figure I gives an example of bisimulable graphs.

The two following graphs are bisimulable.

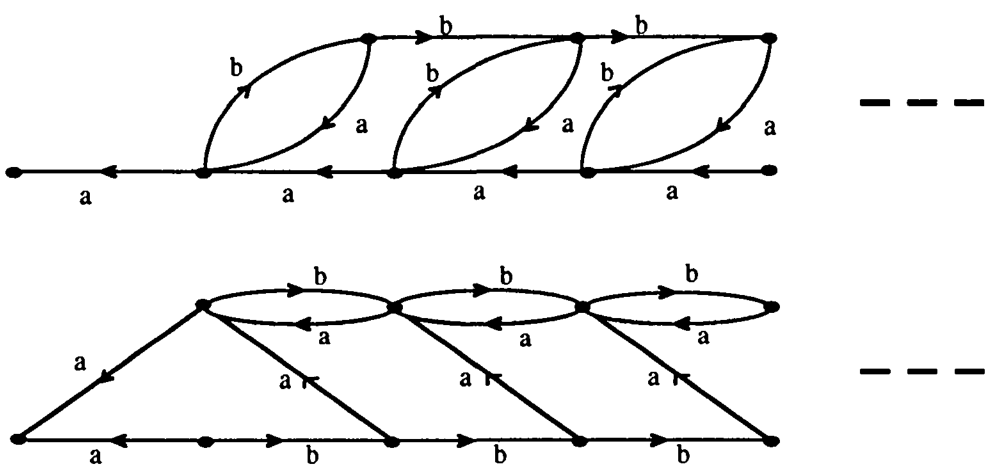


Fig. I . Bisimulable graphs .

The class of graph bisimulations is closed for inverse, union (finite or not) and composition operations. A *self-bisimulation* of a graph is a bisimulation of the graph with itself. The greatest (for inclusion) self-bisimulation is then the union of all self-bisimulations of the graph.
A *reduction* is a bisimulation which is also a function. Reductions are special cases of surjective graph morphisms.

Proposition 3.1. *A reduction* h *from a graph* G *to a graph* H *is a total function from the vertices of* G *onto the vertices of* H *satisfying the two conditions :*

(i) h *is a morphism* : $(s \xrightarrow{f} t) \in G \quad \Rightarrow \quad (h(s) \xrightarrow{f} h(t)) \in H$

(ii) $(h(s) \xrightarrow{f} t') \in H \quad \Rightarrow \quad \exists\, (s \xrightarrow{f} t) \in G\,,\ h(t) = t'$.

Note that there exist non isomorphic graphs which are inter-reducible. For any graph G and for any binary relation R on the vertices of G , G/R denotes the *quotient* of G by R defined as

$G/R \ = \ \{\ R(s) \xrightarrow{f} R(t) \mid (s \xrightarrow{f} t) \in G\ \}$ where R(s) is the set of vertices t satisfying s R t .
It is easily shown that graph reductions are a particular case of quotients.

Proposition 3.2. *Let* h *be a reduction* $G \underline{\leftrightarrow} H$, *then* H *is isomorphic to* G/Ker(h) *where* $Ker(R) = \{\ (u,v) \mid \exists\ w\ ,\ u\,R\,w\ \wedge\ v\,R\,w\ \}$ *is the kernel of a binary relation* R .

Thus, reduction kernels coincide with those self-bisimulations which happen to be equivalences ; this is why reduction kernels are called *congruences* in the sequel. Notice that the greatest self-bisimulation of a graph is always its greatest congruence.

Definition. The *canonical graph* Min(G) of a graph G is the quotient of G by its greatest self-bisimulation.

A graph is *canonical* if it is isomorphic to its canonical graph. Figure J gives an example of canonical graph.

The canonical graphs of the graphs of figure I are isomorphic and are represented by

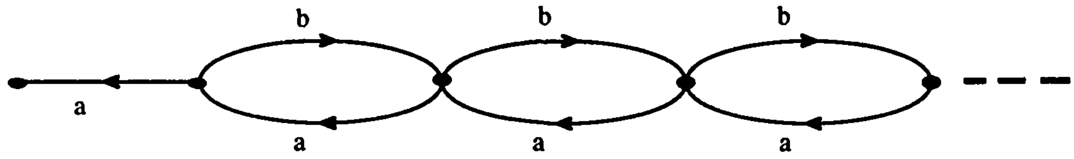
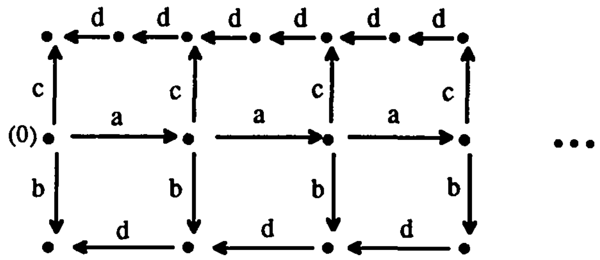


Fig. J . Canonical graph .

Thus, the canonical graph Min(G) of a graph G is the unique (up to isomorphim) graph obtained from G by reduction, and which is irreducible. In consequence, a graph is bisimilar to its canonical graph and more generally, two graphs are bisimilar if and only if they have the same canonical graph (up to isomorphism) and if and only if they reduce to a same graph.

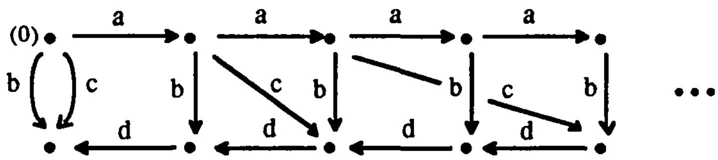
We are now going to study the canonical graphs of the pushdown transition graphs. The canonical graph of a pushdown transition graph is not always a pushdown transition graph. Figure K is an example given to us by G. Sénizergues.

Let us consider the following pattern graph with root 0 and finite degree.



From corollary 1.4 and proposition 2.1, it is a pushdown transition graph, and not alphabetic.

On the contrary, its canonical graph, drawn hereafter



is not a pattern graph, hence not a pushdown transition graph.

Fig. K . Canonical graph of a pushdown transition graph.

On the other hand, we want to show that the class of alphabetic graphs is closed for the quotient by greatest self-bisimulation.

Conjecture 3.3 . *The canonical graph of an alphabetic graph is also alphabetic.*

Henceforth $G = P(R,r)$ is a transition graph of an alphabetic system R on (X,Σ). To establish the conjecture, i.e. $Min(G)$ is alphabetic, we limit ourselves to the case where G has a *terminal* coroot c , i.e. c is source of no arc ; then c is unique.

Possibly using the transformation in proposition 2.10 , we can suppose that the alphabetic system R is *reduced,* that is to say the empty word ε can be obtained from any letter A of X , i.e. ε is a vertex of $P(R,A)$; in particular $c = \varepsilon$. The valuation of the vertices of G is extended to the words α of X^* : $v(\alpha)$ is the shortest length minus one of the paths in $P(R,\alpha)$ from α to ε.

We mean to construct an alphabetic system S and an axiom s whose prefix transition graph $P(S,s)$ is isomorphic to Min(G) . To do this, we define the bisimulation $\underline{\leftrightarrow}$ as a binary relation on X^* , as follows :

$$\alpha \underline{\leftrightarrow} \beta \text{ if there is a bisimulation } U \text{ of } P(R,\alpha) \text{ on } P(R,\beta) \text{ such that } \alpha\, U\, \beta .$$

This relation satisfies the following property : for all $\alpha , \beta \in X^*$,

$$\alpha \underline{\leftrightarrow} \beta \wedge \gamma \underline{\leftrightarrow} \delta \quad \text{iff} \quad \alpha\gamma \underline{\leftrightarrow} \beta\delta \wedge v(\alpha) = v(\beta) .$$

So, the bisimulation relation $\underline{\leftrightarrow}$ is a simplifiable congruence which preserves the valuation.

Now, we are going to construct a finite *generating system* U of the bisimulation, i.e. U is a finite binary relation on X^* such that the smallest congruence $\overset{*}{\underset{U}{\longleftrightarrow}}$ containing U is the bisimulation relation $\underline{\leftrightarrow}$. So, we adjust the notion of self-proving relation [Co 83] to the bisimulation relation.

Definition. A binary relation U on X^* is *self-bisimulable* if

(i) $\alpha\, U\, \beta \wedge \alpha \xrightarrow{f} \alpha' \quad \Rightarrow \quad \exists\ (\beta \xrightarrow{f} \beta')\ ,\ \alpha' \overset{*}{\underset{U}{\longleftrightarrow}} \beta'$

(ii) $\alpha\, U\, \beta \wedge \beta \xrightarrow{f} \beta' \quad \Rightarrow \quad \exists\ (\alpha \xrightarrow{f} \alpha')\ ,\ \alpha' \overset{*}{\underset{U}{\longleftrightarrow}} \beta'$.

Stated otherwise, a relation U is self-bisimulable if the smallest congruence containing U can bisimulate the successor vertices of the pairs of U . Every pair of the smallest congruence containing a self-bisimulable relation is then in bisimulation.

Proposition 3.4 . *If* U *is self-bisimulable then* $\overset{*}{\underset{U}{\longleftrightarrow}}$ *is included in* $\underline{\leftrightarrow}$.

Let us define a constructible class of binary relations on X^* which includes generating systems of the bisimulation.

Definition. A binary relation U on X^* is called *fundamental* if it fulfils the three following conditions :

(i) $\mathrm{Dom}(U) \subseteq X \wedge \mathrm{Im}(U) \subseteq (X - \mathrm{Dom}(U))^*$

(ii) $A\,U\,\alpha \wedge A\,U\,\beta \Rightarrow \alpha = \beta$

(iii) $A\,U\,\alpha \Rightarrow v(A) = v(\alpha)$.

Every fundamental relation U is finite, and a step $\xrightarrow[U]{}$ of rewriting according to U is a relation which is noetherian and confluent (canonical). Moreover, the set of fundamental relations is finite and constructible. It remains to show how to extract one which is a generating system of $\underline{\leftrightarrow}$.

Proposition 3.5 . *Every fundamental and self-bisimulable relation, maximal for inclusion, generates the bisimulation.*

This proposition rests on the following property (of cutting) :

$$\alpha\gamma \underline{\leftrightarrow} \beta\delta \wedge v(\alpha) \geq v(\beta) \Rightarrow \exists\,\lambda\,,\ \alpha \underline{\leftrightarrow} \beta\lambda \wedge \lambda\gamma \underline{\leftrightarrow} \delta\,.$$

Let us take a relation U fundamental, self-bisimulable and maximal for the inclusion (the empty relation is fundamental and self-bisimulable). Let $\alpha{\downarrow}U$ be the reduced word from α according to U and irreducible. From proposition 3.5 , we have $\alpha \underline{\leftrightarrow} \beta$ iff $\alpha{\downarrow}U = \beta{\downarrow}U$. So, the bisimulation is decidable. Moreover, the system

$$S = \{\, A \xrightarrow{f} \alpha{\downarrow}U \mid (A \xrightarrow{f} \alpha) \in R \wedge A \notin \mathrm{Dom}(U) \,\}$$

is alphabetic, and $P(S, r{\downarrow}U)$ is isomorphic to $P(R,r)$. Consequently, conjecture 3.3 is established in an effective way, for every alphabetic graph with a terminal coroot [Ca 90 a] .

Theorem 3.6 . *Given a triple* (R,r,c) *of an alphabetic system* R , *an axiom* r *and a terminal coroot* c *of the prefix transition graph* P(R,r) , *one can transform* (R,r,c) *into a pair* (S,s) *where* S *is an alphabetic system and* P(S,s) *is isomorphic to* Min(P(R,r)) .

This theorem gives a positive answer to the question stated in [Ba-Be-Kl 87] . Yet, conjecture 3.3 remains open for the alphabetic graphs without terminal coroot.

Conclusion

We saw (in section 1) that we can transform every pushdown automaton into a graph grammar generating its transition graph.

By transformation of such a grammar, we were able to decide (in section 2) if a transition graph is alphabetic, in which case, we get (in section 3) a grammar of the canonical graph.

We can then decide on the equivalence problem for a subclass of real-time dpda : given two such dpda (R,r,A) and (S,s,B) where A and B are coaccessible finite sets of the respective transition graphs P(R,r) and P(S,s) , we can decide if these graphs are alphabetic, and in that case, we can decide if the dpda recognize the same language with acceptance on A and B respectively.

This work is a first approach to study the canonical graphs of solutions of recursive programs schemes, and in the monadic case, it allows to extract patterns of these graphs from the schemes.

References

Ba-Be Kl 87 — J.C.M. Baeten, J.A. Bergstra, J.W. Klop 'Decidability of bisimulation equivalence for processes generating context-free languages', LNCS 259, pp 94-111.

Ba 89 — M. Bauderon 'On systems of equations defining infinite graphs', LNCS 344, pp 54-73.

Ca 90 a — D. Caucal 'Graphes canoniques de graphes algébriques', Report Inria 872, will be published in Rairo.

Ca 90 b — D. Caucal 'On the regular structure of prefix rewriting', Caap 90, LNCS 431.

Co 83 — B. Courcelle 'An axiomatic approach to the KH algorithms', Math. Syst. Theory 16, pp 191-231.

Co 89 a — B. Courcelle 'The monadic second-order logic of graphs, II : Infinite graphs of bounded width', Math. Syst. Theory 21, pp 187-222.

Co 89 b — B. Courcelle 'The definability of equational graphs in monadic second-order logic', Icalp, LNCS 372, pp 207-221.

Ha-Kr 87 — A. Habel, H.J. Kreowski 'Some structural aspects of hypergraph languages generated by hyperedge replacement', LNCS 247, pp 207-219.

Mu-Sc 85 — D. Muller, P. Schupp 'The theory of ends, pushdown automata, and second order logic', TCS 37, pp 51-75.

Pa 81 — D. Park 'Concurrency and automata on infinite sequences', LNCS 104, pp 167-183.

Appendix

To prove proposition 2.4 , we give an another but similar proof of theorem 1.3 .

Theorem 1.3 . *Any pair* (G,M) *of a deterministic graph grammar* G *and an axiom* $M = fs_1 \ldots s_n$ *, such that* $G^{\omega}(M)$ *has finite degree and root* s_1 *, may be effectively transformed into a pair* (R,r) *, of a word rewriting system* R *and an axiom* r *, such that the corresponding graphs* $G^{\omega}(M)$ *and* P(R,r) *are isomorphic.*

Proof.

From lemma 2.1 of [Ca 90 b] , we can assume that G is a connected grammar in standard form, i.e.

G is *proper* : for all rule (X,H) of G , every vertex of X is a vertex of a terminal arc of H

G is in *Greibach form* : for all rule (X,H) of G , the vertices of every non-terminal hyperarc of H are disjoints from the X's ones,

G has *separated outputs* : for all rule (X,H) of G , two non-terminal hyperarcs of H have no common vertex, and every non-terminal hyperarc of H has distinct vertices.

Let us recall that G is *connected* if for all hyperarc X in Dom(G) , $G^{\omega}(X)$ is connected.

After a possible renaming of labels (and possibly adding new rules), we suppose further that two non-terminal hyperarcs in a right hand member of G have different labels. Finally and after a possible renaming of vertices, we may assume that the right member hypergraphs of the rules have no common vertex.

Let N be the set of non-terminals of G , and V be the set of vertices of G . With each rule (X,H) of G , we associate a total function p_X from V_H to $V \cup V.(N{\times}N)$, which is the identity on the set of vertices of H which do not belong to non-terminal hyperarcs of H . For any vertex s of a non-terminal hyperarc Y of H , we have $p_X(s) = T(i).(T(1),X(1))$ where T is the non-terminal hyperarc in the domain of G with the same label as Y , and i is the place of s in Y , i.e.

$p_X(s) = s$ for $s \in V_H$ such that $s \notin V_J$ for all $J \in H$ and $J(1) \in N$

$p_X(s) = T(i).(Y(1),X(1))$ if there exist $Y \in H$ and $T \in Dom(G)$ such that $Y(i) = s$ and $T(1) = Y(1)$.

Since G has separated outputs, p_X is well defined.

Let R(G) be the rewriting system on $(N{\times}N) \cup V$ defined by

$$R(G) = \{ p_X(s) \xrightarrow{a} p_X(t) \mid \exists H, (X,H) \in G \wedge (s \xrightarrow{a} t) \in H \wedge a \notin N \}.$$

Then R(G) is a normal system (for all rule $u \xrightarrow{a} v$ of R(G), both u and v have length strictly smaller than 3) and an ε-free system (for all rule $u \xrightarrow{a} v$ of R(G), both u and v are nonempty). Furthermore $P(R(G),s_1)$ is isomorphic to $G^{\omega}(M)$. ◆

Applied to the grammar of figure B, this construction gives the following normal rewriting system R :

$$R = \{ 1 \xrightarrow{a} 3(A,A) \ , \ 1 \xrightarrow{a} 2 \ , \ 2 \xrightarrow{a} 1(A,A) \ , \ 2(A,A) \xrightarrow{b} 3 \}.$$

and P(R,1) is isomorphic to the graph of figure C.

Proposition 2.1 . *Every alphabetic graph is of finite multiplicity.*

Proof. Let us suppose there exists an alphabetic graph G with two vertices t and t' having an infinity of multiple starts. So there exists a multiple start Aα for t and t' with A a letter and $|\alpha| > \max(|t|,|t'|)$. Let u and v be two paths of source Aα, respective targets t and t' and without internal common vertex. Then there exists $1 < i < |u|$ and $1 < j < |v|$ such that $u(i) = \alpha = v(j)$ which is a contradiction. ◆

Proposition 2.4 . *Any pair* (H,M) *of a grammar* H *with constrained returns and axiom* $M = fs_1...s_n$ *such that* $H^{\omega}(M)$ *has finite degree and root* s_1, *may effectively be transformed into a pair* (R,r) *of an alphabetic system* R *with axiom* r, *such that the graphs* $H^{\omega}(M)$ *and* P(R,r) *are isomorphic.*

Proof.

As for the proof of theorem 1.3 , we may assume that H is a connected grammar in standard form, that two non-terminal hyperarcs in a right hand member of H have different labels, and that the right member hypergraphs of the rules have no common vertex.

Furthermore and after a possible renaming of labels (and adding rules), we suppose that H is a grammar whose constrained returns are the first vertices of the non-terminal hyperarcs, that is to say for every non-terminal hyperarc $gx_1 \ldots x_m$ of all right member hypergraph K of H, if x_i is a source of a terminal arc of K then $i = 1$.

The construction in the proof of theorem 1.3 gives a system R(H) normal and ε-free such that $P(R(H),s_1)$ is equal to $H^{\omega}(M)$. If H has constrained returns, R(H) has a peculiar form which allows the construction of an alphabetic system S to be constructed now.

To transform R(H) into an alphabetic system S with the same transition graph, we delete in the rules of R(H) the vertices in $V_0 = \{ X(2) \mid X \in Dom(H) \}$, i.e. the first vertex of the left member hyperarcs of H. To do so, we consider a new symbol A, belonging neither to the set N of non-terminals of H nor to the set V of the vertices of H. We denote by h the alphabetic morphism on $V.(N \times N)^*$ erasing on V_0 ($h(s) = \varepsilon$ for all s in V_0) and otherwise restricted to the identity ($h(s) = s$ for all s in $V - V_0 \cup (N \times N)$). The system S on $V \cup \{A\} \cup (N \times N)$ defined by

$$S = \{ h(u) \xrightarrow{a} h(v) \mid (u \xrightarrow{a} v) \in R(H) \wedge u \notin V_0 \}$$
$$\cup \{ (X(1),B) \xrightarrow{a} h(v).(X(1),B) \mid X \in Dom(H) \wedge (X(2) \xrightarrow{a} v) \in R(H) \wedge B \in N \}$$
$$\cup \{ A \xrightarrow{a} h(v).A \mid (s_1 \xrightarrow{a} v) \in R(H) \} .$$

is alphabetic, and $P(S,A) = h(P(R(H),s_1A)) = \{ h(u) \xrightarrow{a} h(v) \mid (u \xrightarrow{a} v) \in P(R(H),s_1A) \}$.

So P(S,A) is isomorphic to $P(R(H),s_1)$ and so to $H^{\omega}(M)$. ◆

Applied to the grammar $H = \{ (X1,\{a12,Y2\}) , (Y3,\{b34,c43,X4\}) \}$ and the hyperarc X1, the proof of theorem 1.3 gives the following normal and ε-free system :

$$R = \{ 1 \xrightarrow{a} 3(Y,X) \ , \ 3 \xrightarrow{b} 1(X,Y) \ , \ 1(X,Y) \xrightarrow{c} 3 \} .$$

such that $P(R,1) = H^{\omega}(X_1)$. From proof of proposition 2.4, P(R,1) is isomorphic to P(S,A) where S is the following alphabetic system :

$$S = \{ A \xrightarrow{a} (Y,X)A \ , \ (X,X) \xrightarrow{a} (Y,X)(X,X) \ , \ (X,Y) \xrightarrow{a} (Y,X)(X,Y) \}$$
$$\cup \{ (Y,X) \xrightarrow{b} (X,Y)(Y,X) \ , \ (Y,Y) \xrightarrow{b} (X,Y)(Y,Y) \cup \{ (X,Y) \xrightarrow{c} \varepsilon \} .$$

The proof of proposition 2.5 rests on lemma A.5 . Let us recall that the valuation v(s) of a vertex s of G with coroot c , is the minimum number of arcs needed to reach c from s . We write u[i,j] instead of u(i)...u(j) where u is a path and $1 \le i \le j \le |u|$.

Lemme A.5 . *In a corooted graph, let* s *be a multiple start for the pair* $\{t_1,t_2\}$, *the three vertices being distinct. Then* s *is also a multiple start for a vertex* t *whose valuation is less than the greatest valuations of* t_1 *and* t_2 , *i.e.* $v(t) < \max\{v(t_1),v(t_2)\}$.

Proof.

Let $t_1 \neq t_2$ be vertices in G and u' a path from t_1 to the coroot c of minimal length. So $|u'| = v(t_1) + 1$ and $v(u'(i+1)) = v(u'(i)) - 1$ for all $1 \le i < |u'|$. In the same way, let v' be a path from t_2 to c . Let us take the first vertex u'(p) = v'(q) of u' common to v' , and put down u = u'[1,p] and v = v'[1,q] . Let s be a multiple start of $\{t_1,t_2\}$ distinct from t_1 and t_2 . To prove this lemma, it suffices to show that s is a multiple start of a vertex of u distinct from t_1 , or a vertex of v distinct from t_2 .

By definition of a multiple start, there exist two elementary paths x and y with the same source s , respective targets t_1 and t_2 , without internal common vertex.

If u and y have common vertices, we take y(i) = u(m) the first vertex of y common to u , and u(n) = x(j) the last vertex of u[1,m] common to x . Then x[1,j].u[n+1,m] and y[1,i] are two elementary paths of source s and target u(m) , without internal common vertex. So s is a multiple start of u(m) . As $s \neq t_1$, we have $u(m) = y(i) \neq t_1$.

Also, if v and x have a common vertex then s is a multiple start of a vertex of v , distinct from t_2 .

Now, we have only to consider the case where neither u and y , nor v and x , have a common vertex. In this case, let u(i) = x(m) the last vertex of u common to x , and v(j) = y(n) the last vertex of v common to y . Then x[1,m].u[i+1,|u|] and y[1,n].v[j+1,|v|] are elementary paths of the same source s , the same target $u(|u|) = v(|v|)$ and without internal common vertex.

So s is a multiple start of $u(|u|) = v(|v|)$, distinct from t_1 or t_2 because $t_1 \neq t_2$. ♦

Proposition 2.5 . *A corooted graph has a finite multiplicity if and only if every vertex has only a finite number of multiple starts.*

Proof.

The sufficient condition follows from the definition of the finite multiplicity. By contraposition, let us state the necessary condition. From lemma A.5, if a pair $\{t,t'\}$ of distinct vertices has an infinite number of multiple starts, then an infinite number of them are multiple starts for a vertex s such that $v(s) < \max(v(t),v(t'))$, because there is only a finite number of such vertices. So, if the graph is not of finite multiplicity then it has a vertex with an infinity of multiple starts. ♦

Proposition 2.6 . *Let us consider a graph with coroot* c . *The codomination relation* $\leq$ *is an order relation with least element* c , *and strictly growing valuations, i.e.*

$$s < t \Rightarrow v(s) < v(t) .$$

The direct codomination relation $\prec$ *defines a tree structure on the graph vertices, i.e.*

$$(s \leq r \wedge t \leq r) \Rightarrow (s \leq t \vee t \leq s) .$$

This tree, called codomination tree, is of finite degree if and only if the graph has finite multiplicity.

Proof.

Let us show that the valuation is strictly growing for the codomination relation. Let $s < t$. Let us consider a minimal path u from t to c. So there exists $1 < i \leq |u|$ such that $u(i) = s$.

Hence $v(s) = |u| - i < |u| - 1 = v(t)$.

The antisymmetry of the codomination follows from this. The reflexivity and transitivity are evident. As c is the only vertex of valuation zero of the graph, and every path of target c goes through c , the coroot c is the least element for $\leq$.

It remains to prove that $\prec$ defines a tree structure. Let us suppose we have $s \leq r$, $t \leq r$ and $v(s) \leq$

$v(t)$, and let us show that $s \leq t$. Let u be a path of minimum length from r to c. As $t \leq r$, this path goes through t, i.e. $u(i) = t$ for some i. By minimality of $|u|$, $v(u(j)) > v(u(i)) = v(t) \geq v(s)$ for every $1 \leq j < i$; hence s is not a vertex of $u[1,i-1]$. Let v be any path from t to c. Then $u[1,i-1].v$ is a path from r to c going through s because $s \leq r$.
Then v goes through s. Hence $s \leq t$. ◆

Before establishing lemma 2.7, let us point out the opposite passage of lemma A.5, i.e. from a multiple start of a vertex to a multiple start of a pair of vertices. Let $Dm\{s,t\}$ be the set of the multiple starts of $\{s,t\}$.

Lemma A.7. *Every multiple start* s *of a vertex* t *is a multiple start of* $\{t,t'\}$ *where* t' *is the target of a path of source* s *which doesn't go through* t.

Proof.

Let s be a multiple start of t, and w a path from s to t' not going through t. There are two elementary paths u and v of source s, target t and without internal common vertex. Let $w(i)$ be the last vertex of w common to u or v. By symmetry of u and v, we suppose $w(i) = v(j)$. Then u and $v[1,j]w[i+1,|w|]$ are paths from s to t and t' respectively, without common vertex besides s (because $v(j) = w(i) \neq t$). So s is a multiple start of $\{t,t'\}$. ◆

Lemma 2.7. *The direct codominator of* $t \neq c$ *is the strict codominator of* t *with* t *as multiple start, i.e.* $s \prec t$ *iff* $s < t \wedge t \in Dm\{s\}$.

It is also the vertex of minimum valuation with t *as multiple start, i.e.*

$$s \prec t \quad \textit{iff} \quad s \in E_t \wedge \forall r \in E_t,\ v(s) \leq v(r).$$

Proof.

i) If $s \prec t$ then by definition $s < t$. So, there is a minimal path u from t to s. For every path v from t to c, the least integer $j > 1$ such that $v(j)$ be a vertex of u does exist, and we write $u(i_v) = v(j)$. Let i be the greatest i_v for the set of the paths v from t to c. So $s \leq u(i) < t$ and as $s \prec t$, we have $s = u(i)$, hence $t \in Dm\{s\}$; this ends the proof of the necessary condition of the first property.

ii) This condition is sufficient. Suppose $s < r \leq t$ and $t \in Dm\{s\}$. We want to prove $r = t$. As t is a multiple start of s , there are two paths u and v from t to s without internal common vertex. Let us take a minimal path sw from s to c . Then uw and vw are paths from t to c going through r because $r \leq t$. As $s < r$, we have $v(s) < v(r)$, therefore r is not a vertex of sw . So r is a common vertex to u and v , distinct from s . So $r = u(1) = v(1) = t$.

iii) The condition in the second property is sufficient : suppose that $s \in E_t$ and $v(s) \leq v(r)$ for all $r \in E_t$, and let us show $s \prec t$. From (ii) , it suffices to establish that $s < t$. By definition of E_t , we have $v(s) < v(t)$ and $t \in Dm\{s\}$. Suppose that there is a path from t to c not going through s ; in particular s is distinct of c . From lemma A.7 , $t \in Dm\{s,c\}$. Moreover, the three vertices s,t,c are distinct. From lemma A.5 , t is a multiple start of a vertex r such that $v(r) < v(s)$. So $r \in E_t$ and $v(r) < v(s)$. This is a contradiction, and $s < t$.

iv) This second condition is necessary : suppose that $s \prec t$. Hence $s < t$ and in particular $v(t) > v(s)$. So, there is a vertex q such that $t \rightarrow q$ and $v(q) = v(t) - 1$. Then $q \in E_t$, hence $E_t \neq \emptyset$. Let $r \in E_t$ be of minimal valuation. From (iii) , $r \prec t$. As $s \prec t$ and from proposition 2.6 , $s = r$. This ends the proof of proposition 2.7 . ♦

Proposition 2.8 . *Given a triple* (R,r,c) *of an automaton* R , *an axiom* r *and a coroot* c *of the accessible transition graph* P(R,r) , *the relations of codomination and strict codomination are decidable.*

Proof.

Let us consider a word rewriting system R and a non-terminal word r whose accessible transition graph P(R,r) has a coroot c . From lemma 1.2 of [Ca 90 b] , we may suppose R normal and ε-free.

i) Let us show that $\prec$ is a decidable relation. From lemma 2.7, to decide that a vertex codomines directly a vertex t , it suffices to construct the finite set E_t , or again to decide if t is a multiple start of a given vertex.

To do so and from theorem 1.2 , we construct a connected graph grammar G of finite degree, in standard form, which generates from a non-terminal hyperarc M the pattern graph P(R,r) by vertices of growing lengths. We establish the existence of a bound b , depending on s and t , such that t is a multiple start of s in the finite graph $G^b(M)$ obtained from M by b parallel rewriting steps, if and only if t is a multiple start of s in P(R,r) .

Let us bring in some notations. Let p = #Dom(G) the number of patterns (or of non-terminals) of the grammar G , and q = max{ |X| - 1 | X ∈ Dom(G) } the maximal arity of the non-terminals of G . The system R being normal, the first two letters of each vertex determine the transitions to apply. Let Pref(s) be the prefix of length min(2,|s|) of a vertex s , and Suff(s) the remaining suffix, i.e.

$$s = Pref(s).Suff(s) \quad \text{where} \quad |Pref(s)| = \min(2,|s|) .$$

Taking a vertex s of P(R,r) , we note $P(R,r)_s$ the connected component of P(R,r) restricted to the vertices of length at least |s| , and containing s . The set V(s) of vertices of $P(R,r)_s$ of length |s| is computable (see the proof of theorem 1.2 of [Ca 90 b]). So, the equivalence ≡ on the set $V_{P(R,r)}$ of vertices of P(R,r) , defined by

$$s \equiv t \quad \text{if} \quad V(s).Suff(s)^{-1} = V(t).Suff(t)^{-1} ,$$

is decidable. Furthermore, if s ≡ t then $P(R,r)_s$ is isomorphic to $P(R,r)_t$, and $\#(V_{P(R,r)}/\equiv) = p$.

For every path u of P(R,r) , let «u» = max{ |u(i)| | 1 ≤ i ≤ |u| } be the maximal length of its vertices, or yet the minimum number (except in a shifting of the minimal length of the vertices in P(R,r)) of parallel rewritings needed to reach all the vertices of u from M . An *elementary path* is a path of distinct vertices. To every path u is associated the elementary path [u] from u(1) to u(|u|) defined recursively by

$$[\varepsilon] = \varepsilon \quad \text{and} \quad [u] = u(1).u[i+1,|u|] \text{ for every path } u \neq \varepsilon \text{ with } i = \max\{ j \mid u(j) = u(1) \} .$$

To decide t ∈ Dm{s} , we begin by giving a proof of the accessibility.

i1) To decide $s \longrightarrow^* t$, it suffices to show that the accessibility of t from s implies the existence of a path u from s to t such that

$$«u» \leq \max(|s|,|t|) + pq(q-1) . \qquad (1)$$

To simplify the notations, we write a = max(|s|,|t|) + pq(q-1) . For every path u from s to t , we show (1) by induction on the number

$n_u = \#\{\ 1 \le i \le |u| \mid |u(i)| > a\ \}$

of vertices of u whose length is greater than a .

Suppose that u is a path from s to t . As $n_{[u]} \le n_u$ and possibly replacing u by [u] , we can suppose that u is elementary. If $n_u = 0$ then (1) is satisfied, else there is $1 \le i_0 \le |u|$ such that $|u(i_0)| > a$. For every $\max(|s|,|t|) \le i \le a$, we denote by

$$m_i = \max\{\ j \mid j \le i_0 \wedge |u(j)| = i\ \}$$

the rank of the last vertex of u before $u(i_0)$ and of length i ,

$$\text{and } M_i = \min\{\ j \mid j \ge i_0 \wedge |u(j)| = i\ \}$$

the rank of the first vertex of u after $u(i_0)$ and of length i .

So $|u(j)| \ge i$ for every $m_i \le j \le M_i$, and consequently, the rewriting being prefix, $Suff(u(m_i))$ is a common suffix to all these u(j) .

There exist $\max(|s|,|t|) \le i < j \le a$ such that the winding between $u(m_i)$ and $u(M_i)$ is long enough to allow the repetition of a couple of vertices $u(m_j)$ and $u(M_j)$ with the same couple of prefixes as $u(m_i)$ and $u(M_i)$, and belonging to the same pattern. In fact, the number a was choosen to that purpose : there are p patterns, and among q elements, there are q(q-1) couples of distinct elements. So, there are $\max(|s|,|t|) \le i < j \le a$ such that

$$(Pref(u(m_i)),Pref(u(M_i))) = (Pref(u(m_j)),Pref(u(M_j))) \quad \text{and} \quad u(m_i) \equiv u(m_j) .$$

Then, this winding can be shortened to the path

$$w = \big((u(m_j).Suff(u(m_j))^{-1}).Suff(u(m_i))\big)\ldots\big((u(M_j).Suff(u(m_j))^{-1}).Suff(u(m_i))\big)$$

from $u(m_i)$ to $u(M_i)$, by deleting the factors $u[m_i,m_j]$ and $u[M_j,M_i]$ of u , diminishing in the same time the number of vertices whose length is greater than a . Thus, $v = u[1,m_i\text{-}1].w.u[M_i\text{+}1,|u|]$ is a path from s to t and $n_v < n_u$. This ends the proof of accessibility.

i2) To decide $t \in Dm\{s\}$, it suffices to show the existence of two paths u and v from t to s , without internal common vertex, and such that

$$\max(«u»,«v») < \max(|s|,|t|) + 3.p.q! . \qquad (2)$$

Suppose that $t \in Dm\{s\}$. First, let us show the existence of two paths u and v from t to s , without internal common vertex, and such that

$$«u» < \max(|s|,|t|) + 2.p.q! . \qquad (3)$$

Let us put down $b = \max(|s|,|t|) + 2.p.q! - 1$ and show (3) by induction on

$n_u = \#\{ 1 \le i \le |u| \mid |u(i)| > b \}$ for every couple (u,v) of paths from t to s, without internal common vertex.

As $t \in Dm\{s\}$, there are two paths u and v from t to s without internal common vertex, such that «u» ≤ «v» . If $n_u = 0$ then (u,v) suits, else there is $1 \le i_0 \le |u|$ such that $|u(i_0)| > b$. Let us recall that $V(s)$ is the set of vertices of $P(R,r)_s$ of length $|s|$. For every $\max(|s|,|t|) \le i \le b$, the ranks of the last vertices of u and v of length i, and before i_0 are :

$$m_i = \max\{ j \le i_0 \mid |u(j)| = i \} \quad \text{and} \quad n_i = \max\{ j \le i_0 \mid |v(j)| = i \} .$$

Let $\quad U_i = \{ (j,k) \mid u(j) \in V(u(m_i)) \wedge |u(j+1)| > i \wedge k = \min\{ n > j \mid |u(n)| = i \} \}$

be the set of the couples of ranks of vertices of u of length i, separated by vertices of lengths greater than i.

Let $\quad V_i = \{ (j,k) \mid v(j) \in V(v(n_i)) \wedge |v(j+1)| > i \wedge k = \min\{ n > j \mid |v(n)| = i \} \}$

the corresponding set for v.

As we saw in (i1) , for $(j,k) \in U_i$ (resp. V_i) , the winding $u[j,k]$ of u (resp. $v[j,k]$) has the common suffix $Suff(u(m_i))$ (resp. $Suff(v(n_i))$) to all its vertices.

Let us show there is $\max(|s|,|t|) \le i < i' \le b$ such that

$$\{ (Pref(u(j)),Pref(u(k)),[u(j)]_\equiv) \mid (j,k) \in U_i\} = \{ (Pref(u(j)),Pref(u(k)),[u(j)]_\equiv) \mid (j,k) \in U_{i'}\}$$

and $\quad \{ (Pref(v(j)),Pref(v(k)),[v(j)]_\equiv) \mid (j,k) \in V_i\} = \{ (Pref(v(j)),Pref(v(k)),[v(j)]_\equiv) \mid (j,k) \in V_{i'}\}$.

To do this, it suffices to show that there is at most $2.q! - 1$ pairs of sets of disjoint couples where disjoint couples have no common element.

Given $2n > 0$ elements, there are $(2n)! / n!$ sets of n disjoint couples. So, among $q \ge 2n$ elements, there are $C_q^{2n} (2n)! / n! = q! / ((q-2n)!\, n!)$ sets of n disjoint couples, then there are $q! [\, 1/((q-2)!\, 1!) + 1/((q-4)!\, 2!) + \ldots] \le q!$ sets of disjoint couples.

So, for a set E of n disjoint couples, it remains less than $(q-2n)!$ sets of disjoint couples and disjoint of the couples in E ; hence there exist at most $q! / ((q-2n)!\, n!).(q-2n)! = q! / n!$ pairs of such sets. Finally and among q elements, there exist at most $q! [\, 1/1! + 1/2! + \ldots] < 2.q!$ pairs of sets of disjoint couples.

Let u' be the path from t to s obtained by replacing in u every factor $u[j,k]$ where $(j,k) \in U_i$ by the copy $\left((u(j').Suff(u(j'))^{-1}).Suff(u(j))\right)\ldots\left((u(k').Suff(u(j'))^{-1}).Suff(u(j))\right)$ of the factor $u[j',k']$

where $(j',k') \in U_{i'}$, $u(j) \equiv u(j')$, $Pref(u(j)) = Pref(u(j'))$ and $Pref(u(k)) = Pref(u(k'))$. We obtain the path v' from v in the same way. So u' and v' are paths from t to s without internal common vertex, with $n_{u'} < n_u$; hence by induction on n_u, we have the property (3).

To prove (2), let us take a couple (u,v) of paths verifying (3). If «v» ≤ «u» then (2) is satisfied, else from (i1), we obtain $«v» - «u» \leq pq(q-1)$.

So $«v» < \max(|s|,|t|) + 2.p.q! + p.q.(q-1) \leq \max(|s|,|t|) + 3.p.q!$. This ends the proof of (2) and of the decision that a vertex is a multiple start for another one. Finally, $\prec$ is decidable.

ii) Let us show that $\leq$ is decidable. Let us take a vertex t. We can construct the finite set $V_t = \{ s \mid v(s) \leq v(t) \}$ of vertices whose valuation is less than or equal to the valuation of t. From (i), we can construct the restriction R_t of $\prec$ on V_t, i.e.

$$R_t = \{ (r,s) \mid r \prec s \wedge r,s \in V_t \},$$

just as its reflexive and transitive closure R_t^*. As $s \leq t$ if and only if $s R_t^* t$, the codomination relation $\leq$ is decidable.

◆

To prove proposition 2.9, we need two intermediate lemmas. First, let us point out that

$$\begin{aligned}
G_n &= \{ s \xrightarrow{f} t \in G \mid \exists i,\ 0 \leq i \leq n-1,\ c \prec^i s \} \cup \{ s \xrightarrow{f} t \in G \mid c \prec^n s \wedge \neg(s \leq t) \}. \\
&= \{ s \xrightarrow{f} t \in G \mid \exists i,\ 0 \leq i \leq n-1, c \prec^i s \wedge s \leq t \} \\
&\quad \cup \{ s \xrightarrow{f} t \in G \mid \exists i,\ 1 \leq i \leq n-1,\ c \prec^i s \wedge \neg(s \leq t) \} \\
&\quad \cup \{ s \xrightarrow{f} t \in G \mid c \prec^n s \wedge \neg(s \leq t) \}. \\
&= \{ s \xrightarrow{f} t \in G \mid \exists i,\ 0 \leq i \leq n-1, c \prec^i s \wedge s \leq t \} \\
&\quad \cup \{ s \xrightarrow{f} t \in G \mid \exists i,\ 1 \leq i \leq n,\ c \prec^i s \wedge \neg(s \leq t) \} \\
&= \bigcup \{ M_r \mid \exists i,\ 0 \leq i \leq n-1,\ c \prec^i r \}.
\end{aligned}$$

Let us give an explicit characterization of $G - G_n$.

Lemma A.9 . *The graph* $G - G_n$ *is the set of arcs whose both vertices are codominated by a same vertex of depth* n *in the codomination tree, i.e.*

$$G - G_n = \{ s \xrightarrow{f} t \in G \mid \exists r,\ c \prec^n r \wedge r \leq s \wedge r \leq t \} \textit{ for all } n \geq 0 .$$

Proof.

i) Let us show the direct inclusion. Let us take an arc $s \xrightarrow{f} t$ of $G - G_n$. By definition of G_n , there is $i \geq n$ such that $c \prec^i s$. So, there is a vertex r with $c \prec^n r \leq s$. If $r = s$ then $r = s \leq t$ by definition of G_n . If $r \neq s$ then $r < s$ and $s \rightarrow t$, hence $r \leq t$. In all cases, $r \leq t$; hence the direct inclusion.

ii) Let us show the opposite inclusion. We take an arc $s \xrightarrow{f} t$ of G such that there is a vertex r with $c \prec^n r$, $r \leq s$ and $r \leq t$. If $r \neq s$ then there is $i > n$ such that $c \prec^i s$, hence $s \xrightarrow{f} t \notin G_n$. If $r = s$ then $c \prec^n s$ and $s \leq t$, hence $s \xrightarrow{f} t \notin G_n$. In both cases, the opposite inclusion is proved. ♦

A characterization of the connected components of $G - G_n$ can be deduced.

Lemme B.9 . *Every connected component* C *of* $G - G_n$ *is the restriction of* G *to the vertices codominated by the unique vertex* r *of* C *such that* $c \prec^n r$.

Proof.

Let $s\uparrow = \{ t \mid s \leq t \}$ be the set of the vertices of G codominated by a vertex s , and $G_{|A} = \{ s \xrightarrow{f} t \in G \mid s,t \in A \}$ the restriction of G to a subset A of vertices of G . From lemma A.9 and for every $n \geq 0$

$$G - G_n = \bigcup \{ G_{|s\uparrow} \mid c \prec^n s \} .$$

From proposition 2.6 , the relation $\prec$ orders the vertices of G in a tree, so $s\uparrow \cap t\uparrow = \emptyset$ for $s \neq t$, $c \prec^n s$ and $c \prec^n t$.

To prove lemma B.9 , it suffices to show that for every vertex s , $G_{|s\uparrow}$ is connected. Let $t \in s\uparrow$ i.e.

$s \leq t$, and let us take a path u from t to c . As $s \leq t$, there is a least integer i such that $u(i) = s$. By minimality of i and by induction on $1 \leq j \leq i$, we have $s \leq u(j)$. Hence u[1,i] is a path in $G_{|s\uparrow}$ from t to s , and $G_{|s\uparrow}$ is connected. ◆

Proposition 2.9 . *Given a triple* (R,r,c) *of an automaton* R , *an axiom* r *and a coroot* c *of the transition graph* P(R,r) , *we can decide if* P(R,r) *is of finite multiplicity, in which case, we construct a codomination grammar of* P(R,r) .

Proof.

The hypothesis being the same as for proposition 2.8 , we take again the notations of part (i) of its proof, that is to say we may suppose R normal and ε-free, and we construct a connected and finite degree graph grammar G in standard form which generates from a hyperarc M of Dom(G) and by vertices of growing length, the pattern graph P(R,r) . Let p = #Dom(G) be the number of non-terminals of G , and $q = \max\{ |X| - 1 \mid X \in Dom(G) \}$ be the maximal arity of the non-terminals of G . We denote by Pref(s) the prefix of length min(2,|s|) of a vertex s , and Suff(s) the remaining suffix, i.e.

$$s = Pref(s).Suff(s) \quad \text{where} \quad |Pref(s)| = \min(2,|s|) .$$

Taking a vertex s of P(R,r) , we note $P(R,r)_s$ the connected component of P(R,r) restricted to the vertices of length at least |s| , and containing s . The set V(s) of vertices of $P(R,r)_s$ of length |s| is computable, and the equivalence ≡ on the set $V_{P(R,r)}$ of vertices of P(R,r) , defined by

$$s \equiv t \quad \text{if} \quad V(s).Suff(s)^{-1} = V(t).Suff(t)^{-1} ,$$

is decidable.

First, let us decide the finite multiplicity of a vertex.

i) Let us show that the finiteness of the set Dm{s} of the multiple starts of a vertex s is decidable. From (i2) in the proof of proposition 2.8 , $t \in Dm\{s\}$ is decidable, and to decide the finiteness of Dm{s} , it suffices to prove the following property (1) :

$$\#Dm\{s\} = \infty \quad \text{iff} \quad \exists\, t \in Dm\{s\} ,\ |s| + 2.p.q! < |t| \leq |s| + 4.p.q! . \qquad (1)$$

To prove the equivalence (1) , let us take a multiple start t of s such that $|t| - |s| > 2.p.q!$. We proceed in the same way as part (i2) of the proof of proposition 2.8 . There are two elementary paths u and v from t to s, without internal common vertex. For every $|s| < i < |t|$, we put down

$$m_i = \min\{ j \mid |u(j)| = i \} \quad \text{and} \quad n_i = \min\{ j \mid |v(j)| = i \},$$

$$U_i = \{ (j,k) \mid u(j) \in V(u(m_i)) \wedge |u(j+1)| > i \wedge k = \min\{ n > j \mid |u(n)| = i \} \}$$

$$\text{and } V_i = \{ (j,k) \mid v(j) \in V(v(n_i)) \wedge |v(j+1)| > i \wedge k = \min\{ n > j \mid |v(n)| = i \} \}.$$

with the same meaning as in (i2) of proposition 2.8.

Leaving apart $u(m_i)$ and $v(n_i)$ among the possible q vertices of length i in a same pattern, we are left with q-2 vertices from which we can obtain at most 2.(q-2)! - 1 pairs of sets of disjoint couples. As $q(q-1)p[2.(q-2)! - 1] < 2.p.q!$, there is $|s| < i < i' < |t|$ such that

$$\text{Pref}(u(m_i)) = \text{Pref}(u(m_{i'})) \quad \text{and} \quad \text{Pref}(v(n_i)) = \text{Pref}(v(n_{i'})).$$

$$\{ (\text{Pref}(u(j)),\text{Pref}(u(k)),[u(j)]_{\equiv}) \mid (j,k) \in U_i \} = \{ (\text{Pref}(u(j)),\text{Pref}(u(k)),[u(j)]_{\equiv}) \mid (j,k) \in U_{i'} \}$$

and $\{ (\text{Pref}(v(j)),\text{Pref}(v(k)),[v(j)]_{\equiv}) \mid (j,k) \in V_i \} = \{ (\text{Pref}(v(j)),\text{Pref}(v(k)),[v(j)]_{\equiv}) \mid (j,k) \in V_{i'} \}$.

Let u' be the path obtained by replacing in u every factor $u[j,k]$ where $(j,k) \in U_i$ by the copy $\big((u(j').\text{Suff}(u(j'))^{-1}).\text{Suff}(u(j))\big)\ldots\big((u(k').\text{Suff}(u(j'))^{-1}).\text{Suff}(u(j))\big)$ of the factor $u[j',k']$ where $(j',k') \in U_{i'}$, $u(j) \equiv u(j')$, $\text{Pref}(u(j)) = \text{Pref}(u(j'))$ and $\text{Pref}(u(k)) = \text{Pref}(u(k'))$, and by replacing the factor $u[1,m_i]$ by the copy

$\big((u(1).\text{Suff}(u(m_{i'}))^{-1}).\text{Suff}(u(m_i))\big)\ldots\big((u(m_{i'}).\text{Suff}(u(m_{i'}))^{-1}).\text{Suff}(u(m_i))\big)$ of $u[1,m_{i'}]$. We construct the path v' from v in the same way. So u' and v' are paths from

$t' = \big((t.\text{Suff}(u(m_{i'}))^{-1}).\text{Suff}(u(m_i))\big)$ to s, without internal common vertex. Hence $t' \in Dm\{s\}$ and $|t| - 2.p.q! < |t'| < |t|$.

So, if $\#Dm\{s\} = \infty$ then there is $t \in Dm\{s\}$ such that $|t| > |s| + 2.p.q!$. Hence, by induction on $|t|$, there is $t \in Dm\{s\}$ such that $|t| > |s| + 2.p.q!$ and $|t| \leq |s| + 4.p.q!$; so, we have the necessary condition of (1).

Conversely, if there is $t \in Dm\{s\}$ such that $|t| > |s| + 2.p.q!$ then we consider the path u" obtained by replacing in u each winding $u[j',k']$ where $(j',k') \in U_{i'}$ by the longest winding $\big((u(j).\text{Suff}(u(j))^{-1}).\text{Suff}(u(j'))\big)\ldots\big((u(k).\text{Suff}(u(j))^{-1}).\text{Suff}(u(j'))\big)$ where $(j,k) \in U_i$, $u(j) \equiv u(j')$, $\text{Pref}(u(j)) = \text{Pref}(u(j'))$ and $\text{Pref}(u(k)) = \text{Pref}(u(k'))$, and by replacing the factor $u[1,m_{i'}]$ by $\big((u(1).\text{Suff}(u(m_i))^{-1}).\text{Suff}(u(m_{i'}))\big)\ldots\big((u(m_i).\text{Suff}(u(m_i))^{-1}).\text{Suff}(u(m_{i'}))\big)$. We construct the path v" accordingly. Then u" and v" are paths from $t'' = \big((t.\text{Suff}(u(m_i))^{-1}).\text{Suff}(u(m_{i'}))\big)$ to s, without internal common vertex. Hence $t'' \in Dm\{s\}$ and $|t''| > |t|$. So and by induction, $Dm\{s\}$ is infinite and the sufficient condition of (1) is proved.

ii) The finite multiplicity of a given vertex being settled, let us decide the finite multiplicity of $P(R,r)$, and in the affirmative, construct a codomination grammar H which generates $P(R,r)$.

From [Mu-Sc 85] or [Co 89], we decide the finiteness of the multiplicity of $P(R,r)$ by expressing it as a monadic second order logic formula. Yet, we give here a direct proof from the construction of H. This construction is analogous to the construction of proposition 3.2 in [Ca 90 b].

Let X be the set of non-terminals of R. We define an order $\leq$ on X that we extend lexicographically on X^*, and for all graph C with vertices in X^* and every non-terminal word u, we write

$$C.u^{-1} = \{ fs_1 \ldots s_n \mid f(s_1u)\ldots(s_nu) \in C \} \text{ the right quotient of } C \text{ by } u,$$

and $\quad s_C$ the greatest common suffix of $\{ Suff(s) \mid s \in V_C \}$.

The grammar H to be constructed must generate from a non-terminal hyperarc, in n steps of parallel rewritings, the graph $G_n = \bigcup\{ M_r \mid \exists\, i,\ 0 \leq i \leq n\text{-}1 \ \wedge\ c \overset{i}{\prec} r \}$ as set of terminal arcs. A non-terminal of H will be a couple (P,Q) where P is a finite set of terminal arcs with vertices in X^*, and Q is a subset of vertices of P.

Let $n \geq 1$. We will determine a set N_n of non-terminal hyperarcs allowing the generation of the graph $P(R,r) - G_n$ according to H. To this aim, we determine the connected components $C_1,\ldots,C_d$ of the set $\{ (s \overset{f}{\longrightarrow} t) \in P(R,r) - G_n \mid s \in V_{G_n} \vee t \in V_{G_n}\}$ of terminal arcs of $P(R,r) - G_n$ having a vertex in G_n

The hypergraph N_n is defined by

$$N_n = \{ (C_i.(s_{C_i})^{-1} , (V_{C_i} \cap V_{G_n}).(s_{C_i})^{-1}).u_{i,1}\ldots u_{i,q_i} \mid 1 \leq i \leq d$$
$$\wedge\ \{u_{i,1},\ldots,u_{i,q_i}\} = V_{C_i} \cap V_{G_n} \ \wedge\ \forall j,\ 1 \leq j < q_i,\ u_{i,j} < u_{i,j+1} \}.$$

The grammar H we look for, is defined as the union of a sequence $(H_n)_{n\geq 1}$ which is inductively constructed as follows :

$$H_1 = \{ ((\emptyset,\emptyset)c, G_1 \cup N_1) \}$$

and $\quad H_{n+1} = \{ (X,C) \mid X \in N_n \ \wedge\ X(1) \notin P_n \ \wedge\ C$ is the connected component of $(G_{n+1} - G_n) \cup N_{n+1}$ having the vertices of $X \}$, where P_n is the set of the non-terminals of $H_1, \ldots, H_n$.

So, and by induction on $n \geq 1$, $H_1 \cup \ldots \cup H_n$ generates from $(\emptyset,\emptyset)c$ in n steps of parallel rewritings, the graph $G_n \cup N_n$.

The graph $P(R,r)$ is not of finite multiplicity iff there is an integer n for which G_n has a vertex that is not of finite multiplicity. So and from (i) , the non finiteness of the multiplicity of $P(R,r)$ is semi-decidable. But, the finite multiplicity of $P(R,r)$ is decidable as $P(R,r)$ is of finite multiplicity iff there is an n such that $H_n = \emptyset$, then the grammar $H = H_1 \cup \ldots \cup H_{n-1}$ is a codomination grammar of $P(R,r)$. In fact, the sufficient condition is straightforward. Conversely, if $P(R,r)$ is of finite multiplicity, it suffices to show that there exists only a finite number of possible non-terminals for H . Again, it suffices to find a bound on $|s| - |t|$ for every vertices s and t common to C and G_n , for every n and every connected component C of $P(R,r) - G_n$. We can restrict ourselves to the integers $n \geq n_0$ for n_0 such that for every vertex s of $P(R,r) - G_{n_0}$, we have $|s| \geq |c|$. From lemma B.9 , it suffices to prove the following step (iii) .

iii) Suppose that $P(R,r)$ be of finite multiplicity, and let us show the existence of a bound b such that for every vertex s of length $\geq |c|$, and for every vertex t of length $\geq |c|$ common to $P(R,r)_{|s\uparrow}$ and $P(R,r) - P(R,r)_{|s\uparrow}$, we have $-b \leq |s| - |t| \leq b$.

First, let us show the following property (2) :

$$s \leq t \wedge |t| \geq |c| \quad \Rightarrow \quad |s| - |t| \leq pq(q-1) . \qquad (2)$$

The proof of (2) is similar to the proof of (i1) of proposition 2.8 . Suppose that $s \leq t$ with $|t| \geq |c|$, and let u be a path from t to c of minimal length. There is i_0 such that $u(i_0) = s$. Suppose that $|s| - |t| > pq(q-1)$ and let us show that this leads to a contradiction. For every $|t| \leq i < |s|$, we put down

$$m_i = \max\{ j \mid j \leq i_0 \wedge |u(j)| = i \}$$

$$\text{and} \qquad M_i = \min\{ j \mid j \geq i_0 \wedge |u(j)| = i \} .$$

As $|t| \geq |c|$, the integers M_i do exist. As u is elementary, there is $|t| \leq i < j < |s|$ such that

$$(Pref(u(m_i)),Pref(u(M_i))) = (Pref(u(m_j)),Pref(u(M_j))) \quad \text{and} \quad u(m_i) \equiv u(m_j) .$$

So $w = \big((u(m_j).Suff(u(m_j))^{-1}).Suff(u(m_i))\big)\ldots\big((u(M_j).Suff(u(m_j))^{-1}).Suff(u(m_i))\big)$ is a path from $u(m_i)$ to $u(M_i)$. Then $v = u[1,m_i-1].w.u[M_i+1,|u|]$ is a path from t to c such that $|v| < |u|$; which contradicts the minimality of $|u|$; the property (2) is then proved.

To find the bound b, let us take a vertex s with $|s| \geq |c|$ and a vertex $t \neq s$ common to $P(R,r)_{|s\uparrow}$ and $P(R,r) - P(R,r)_{|s\uparrow}$ and such that $|t| \geq |c|$. As $s \leq t$ and from (2), we have $|s| - |t| \leq pq(q-1)$. It remains to find a lower bound for $|s| - |t|$. We consider the integer n such that $c \prec^{n} s$. By definition of G_n and from the fact that $s \neq t$, there is a vertex x such that $c \prec^{n} x$, $x \longrightarrow t$ and $\neg(x \leq t)$. So $x \neq s$. Let s' be the nearest ancestor of s and x in the codomination tree. Hence $s' \leq s$ and $x \in Dm\{s'\}$. From (1) and (2) and the finite multiplicity of $P(R,r)$, we have

$$|x| \leq |s'| + 2.p.q! \quad \text{and} \quad |s'| - |s| \leq pq(q-1).$$

As $x \longrightarrow t$ and R is a normal system, we have $|t| \leq |x| + 1$. Finally,

$$|t| - |s| \leq 2.p.q! + pq(q-1) + 1$$

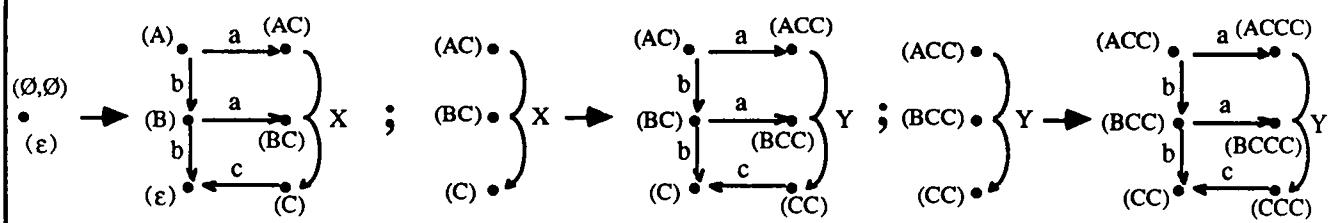
and the bound $b = 2.p.q! + pq(q-1) + 1$ suits. ♦

The construction in the proof of proposition 2.9 is illustrated in figure L.

Let us consider the alphabetic system of figure E whose its codomination tree is represented in figure G.

We take $A < B < C$.

The construction in the proof of proposition 2.9 gives the following codomination grammar :



where $X = (E,F)$ and $Y = (EC,FC)$

with $E = \{ AC \xrightarrow{a} ACC, AC \xrightarrow{b} BC, BC \xrightarrow{a} BCC, BC \xrightarrow{b} C, CC \xrightarrow{c} C \}$ and $F = \{AC,BC,C\}$.

Fig. L . Extraction of a codomination grammar from a corooted and accessible prefix transition graph of finite multiplicity .

LISTE DES DERNIERES PUBLICATIONS INTERNES IRISA

PI 542 A NEW APPROACH TO VISUAL SERVOING IN ROBOTICS
Bernard ESPIAU, François CHAUMETTE, Patrick RIVES
Juillet 1990, 44 Pages.

PI 543 SIMPLE DISTRIBUTED SOLUTIONS TO THE READERS-WRITERS PROBLEM
Michel RAYNAL
Juillet 1990, 10 Pages.

PI 544 IMPLEMENTATION AND EVALUATION OF DISTRIBUTED SYNCHRONIZATION ON A DISTRIBUTED MEMORY PARALLEL MACHINE
André COUVERT, René PEDRONO, Michel RAYNAL
Juillet 1990, 14 Pages.

PI 545 ESTIMATION OF NETWORK RELIABILITY ON A PARALLEL MACHINE BY MEANS OF A MONTE CARLO TECHNIQUE
Mohamed EL KHADIRI, Raymond MARIE, Gerardo RUBINO
Août 1990, 20 Pages.

PI 546 LIMIT THEOREMS FOR MIXING PROCESSES
Bernard DELYON
Septembre 1990, 22 Pages.

PI 547 PERFORMANCES DES COMMUNICATIONS SUR LE T-NODE
Frédéric GUIDEC
Septembre 1990, 38 Pages.

PI 548 LES PREDICATS COLLECTIFS : UN MOYEN D'EXPRESSION DU CONTROLE DU PARALLELISME *OU* EN PROLOG
René QUINIOU, Laurent TRILLING
Septembre 1990, 34 Pages.

PI 549 NORMALISATION SOUS HYPOTHESE D'ABSENCE DE LIEN APPLICATION AU CAS NOMINAL
François DAUDE
Septembre 1990, 42 Pages.

PI 550 MULTISCALE SIGNAL PROCESSING : FROM QMF TO WAVELETS
Albert BENVENISTE
Septembre 1990, 28 Pages.

PI 551 ON THE TRANSITION GRAPHS OF AUTOMATA AND GRAMMARS
Didier CAUCAL, Roland MONFORT
Septembre 1990, 46 Pages.

Imprimé en France
par
l'Institut National de Recherche en Informatique et en Automatique

ISSN 0249-6399