BISIMULATION OF CONTEXT-FREE GRAMMARS AND OF PUSHDOWN AUTOMATA *

Didier CAUCAL

IRISA, Campus de Beaulieu, 35042 Rennes, France
E-mail: caucal@irisa.fr

Abstract

We consider the bisimulation on the transition graphs of pushdown automata.

First we give a characterization of bisimulation using the unfolded trees. We recall that the bisimulation is decidable for the subclass of prefix transition graphs of context-free grammars. Furthermore any pushdown transition graph is a regular graph: it can be generated by iterated parallel rewritings of a deterministic graph grammar.

Then we show that the rational restrictions of pushdown transition graphs are exactly and effectively the regular graphs of finite degree. We restrict to the class of graphs which are regular by increasing valuation : the length to a terminal coroot. This subclass strictly contains the prefix transition graphs of reduced context-free grammars. And we show that the bisimulation is decidable on any graph regular by increasing valuation.

Finally we show that the bisimulation on the regular graphs of finite out-degree is equivalent to the bisimulation on the prefix transition graphs of context-free grammars on terms. For the deterministic case, these problems are inter-reducible to the well-known equivalence problem of dpda.

1 Introduction

Bisimulation has been introduced [Pa 81] to test whether two processes have the same behavior. Two vertices of a graph (set of labelled arcs) are bisimilar if they are linked by a simulation whose inverse is also a simulation. For deterministic graphs, two vertices are bisimilar if and only if they have the same unfolded tree. Generally we show that two vertices of a graph are bisimilar if and only if their unfolded trees are obtained from a common tree by pruning identical 'adjacent subtrees'.

It is well known that the equivalence problem is undecidable for the context-free grammars. However the bisimulation is decidable on the prefix transition graph of any reduced context-free grammar [BBK 87]. It is decided by comparing vertices of same valuation (the minimal path-length to the terminal coroot) and by taking into account the regular structure of the graphs. Another method has been introduced [Ca 90 a] : given any

*This paper is an extended version of the article published in CSLI Volume 53, Modal logic and process algebra, pp. 85–106, Stanford, 1995.
This work has been partly supported by ESPRIT BRA 6317 (ASMICS).

reduced context-free grammar, the greatest bisimulation on its prefix transition graph is an equivalence induced by an effective idempotent morphism. It follows that the class of prefix transition graphs of reduced context-free grammars is effectively closed under quotient by the greatest bisimulation. Another consequence is that this greatest bisimulation is an effective Thue congruence : we can construct a finite relation generating the greatest bisimulation. This last method has been extended for any context-free grammar to decide the bisimulation on its prefix transition graph [CHS 92] and on its transition graph [CHM 93].

Although pushdown automata recognize exactly the languages generated by context-free grammars, their (prefix) transition graphs, called pushdown transition graphs, strictly contain the prefix transition graphs of context-free grammars [Ca 90 b]. The decidability of bisimulation on pushdown transition graphs is an open problem. However the bisimulation of deterministic pushdown transition graphs is decidable because this problem is (inter-)reducible to the decidable equivalence of real-time dpda [Ro 86]. Furthermore we know that any pushdown transition graph is effectively a regular graph of finite degree [Ca 90 b], where a regular graph, or equational graph [Co 90], is a graph generated by iterated rewritings according to a deterministic graph grammar. In fact, the pushdown transition graphs and the regular graphs of finite degree have effectively the same connected components and the same accessible subgraphs. More generally, the rational restrictions of pushdown transition graphs are exactly and effectively the regular graphs of finite degree.

We consider the subclass of graphs which are regular by valuation : they can be generated with a deterministic graph grammar, by vertices of increasing valuation from a terminal coroot. This subclass strictly contains the prefix transition graphs of reduced context-free grammars. Furthermore the original method of [BBK 87] can be generalized to the graphs regular by valuation. More precisely to decide the bisimilarity of two vertices of any graph regular by valuation, it suffices to restrict this problem to a finite subgraph of all vertices whose the valuation is bounded by a value computable from the graph grammar and from the valuation of the two vertices. This means that the bisimulation is decidable on any graph regular by valuation.

Finally we consider the context-free grammars on terms. Their accessible prefix transition graphs are more general that the rooted regular graphs of finite out-degree. But these two classes of graphs have the same unfolded trees. So the bisimulation on any regular graph of finite out-degree is decidable if and only if it is decidable on the prefix transition graph of any context-free grammar on terms. Restricted to deterministic graphs, this bisimulation problem is inter-reducible to the equivalence problem of deterministic pushdown automata.

2 Bisimulation

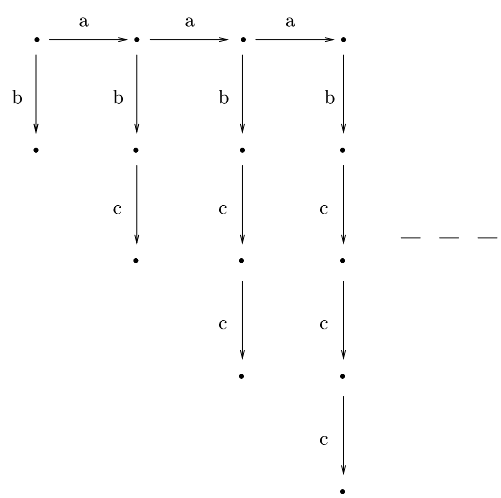
In this section we give some basic properties of (strong) bisimulation. In particular we characterize bisimulation using unfolded trees (Theorem 2.4).

Here, a *graph* G is an arbitrary set of *labelled arcs* (p, a, q) of source p, goal q, and label a. Every arc (p, a, q) may be identified with a *labelled transition* $p \xrightarrow{a} q$. Two graphs are bisimilar if they are linked up by a total relation preserving the branching structure.

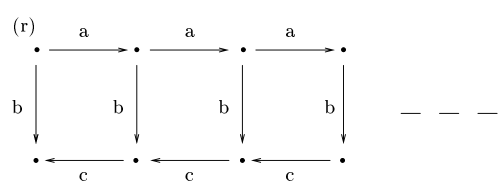
Definition 2.1 [**Pa 81**] Given a binary relation R from the vertices of a graph G into the vertices of a graph H,
R is a *partial simulation* if $p\ R\ q$ and $p \overset{a}{\longrightarrow} p'$ then $q \overset{a}{\longrightarrow} q'$ and $p'\ R\ q'$ for some q';
R is a *simulation* if furthermore R links every vertex p of G: $p\ R\ q$ for some q;
R is a *bisimulation* if R and R^{-1} are simulations, and we write R: $G \sim H$.

Note that the bijective bisimulations are the *isomorphisms*: vertex renamings. The class of bisimulations is closed under inverse, composition and infinite union. A bisimulation on a graph G is a bisimulation from G to G. The union of all the bisimulations on G is the greatest bisimulation $\equiv_G$ on G. Two vertices p, q of a graph G are *bisimilar* if $p \equiv_G q$ or equivalently if there is a bisimulation R on G such that $p\ R\ q$.

We will characterize bisimilar vertices using their unfolded trees. Recall that a *path* in G from a vertex p to a vertex q is either empty (denoted by ϵ) when $p = q$, or is a nonempty finite sequence $(p_0, a_1, p_1) \ldots (p_{n-1}, a_n, p_n)$ of arcs in G with $p_0 = p$ and $p_n = q$. As usual we denote by $p \longrightarrow^* q$ the existence of a path from p to q. The *unfolded tree* $\mathrm{Tree}(G, r)$ of G from a vertex r is the set of arcs $u \overset{a}{\longrightarrow} u(p, a, q)$ where $u(p, a, q)$ is a path from r and $p \overset{a}{\longrightarrow} q$ is an arc of G.
For instance the following tree :



is the unfolded tree of the following graph from vertex r :



Generally a graph T is a *tree* if it has a vertex r such that $\mathrm{Tree}(T, r)$ is isomorphic to T. Equivalently a graph G is a tree if it has a vertex r which is a *root*: every vertex p

is *accessible* from r i.e. $r \longrightarrow^* p$, and furthermore r is target of no arc and every vertex $p \neq r$ is target of a unique arc. A vertex of a tree is also called a *node*.
An *accessible subgraph* of a graph G is the restriction $G/r := \{(p,a,q) \in G \mid r \longrightarrow^* p\}$ of G to the vertices accessible from a vertex r; in particular T/r is the *subtree* of a tree T at node r. We write $G \longrightarrow H$ if there is a functional (injective inverse) bisimulation from G to H, which is called a *reduction*. Note that two graphs are bisimilar if and only if they reduce to a same graph, or equivalently they are reduced from a common graph:

$$G \sim H \quad \text{iff} \quad \exists\, K,\ G \longrightarrow K \longleftarrow H \quad \text{iff} \quad \exists\, K,\ G \longleftarrow K \longrightarrow H.$$

Beware that two inter-reducible graphs are not necessary isomorphic. Given a vertex r of a graph G, the mapping associating its target with every path from r, is a reduction from $\text{Tree}(G,r)$ to G/r. So bisimilar vertices have bisimilar unfolded trees: if $p \equiv_G q$ then $\text{Tree}(G,p) \sim \text{Tree}(G,q)$. The converse is false except if the unfolded trees have bisimilar roots. We bypass this condition using reductions:

$$\begin{aligned} p \equiv_G q \quad &\text{iff} \quad \exists\, T \text{ tree},\ \text{Tree}(G,p) \longrightarrow T \longleftarrow \text{Tree}(G,q) \qquad (*)\\ &\text{iff} \quad \exists\, T \text{ tree},\ \text{Tree}(G,p) \longleftarrow T \longrightarrow \text{Tree}(G,q). \end{aligned}$$

Let us restrict $(*)$ to *deterministic* graphs: distinct arcs with the same label have distinct sources, i.e. if $r \xrightarrow{a} p$ and $r \xrightarrow{a} q$ then $p = q$.

Corollary 2.2 *Given a deterministic graph G,*

$$p \equiv_G q \quad \textit{iff} \quad \text{Tree}(G,p) \textit{ is isomorphic to } \text{Tree}(G,q).$$

To extend Corollary 2.2 to arbitrary graphs, we define a basic operation on trees. Given a tree T and a node q distinct of the root, we define the *partial tree* $T_q := T/q \cup \{(p,a,q)\}$ of T at q, where $p \xrightarrow{a} q$ is the arc of T of goal q. Two partial trees are *adjacent* if they have the same root. A basic reduction on trees consists of removing redundant adjacent partial trees.

Definition 2.3 A tree S is *pruned* to a tree T and we write $S \geq T$, if there is a set P of nodes of S with the following two conditions:
(i) For every p in P there is q not in P such that S_p and S_q are isomorphic adjacent partial trees,
(ii) T is isomorphic to the restriction $S - \bigcup\{S_p \mid p \in P\}$ of S to the nodes not accessible from P.

Note that a deterministic tree S is irreducible according to the pruning relation : $S \geq T$ implies S and T are isomorphic. Furthermore $\geq$ is a reduction on trees : $S \geq T$ implies $S \longrightarrow T$. So vertices are bisimilar when their unfolded trees are obtained from each other by iteratively adding and deleting identical adjacent partial trees. We show that the converse is true: the unfolded trees are obtained by pruning a common tree.

Theorem 2.4 *We have the following equivalences:*

$p \equiv_G q$ *iff* $\text{Tree}(G,p)\ (\geq \cup \leq)^*\ \text{Tree}(G,q)$

iff $\text{Tree}(G,p) \leq T \geq \text{Tree}(G,q)$ *for some tree* T.

Beware that bisimilar vertices may have unfolded trees not pruning to a same tree. This ends the fact that bisimulation preserves branching structure.

3 Bisimulation of context-free grammars

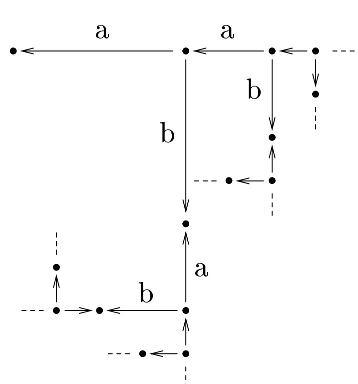
In this section, we recall and refine decision results on bisimulation of context-free grammars (Theorems 3.2 and 3.3).

We consider the decidability of bisimulation. Given an arbitrary finite graph G, the decision of $\equiv_G$ is obvious. Any binary relation R on the vertices of G is finite, so we can decide whether R is a bisimulation. Furthermore the set of binary relations on the vertices of G is finite, hence we can construct the greatest bisimulation $\equiv_G$ on G. In fact we can decide $\equiv_G$ in $O(m.logn)$ [PT 87] where $m := \#G$ is the number of arcs of G (its cardinality) and $n \leq 2m$ is the number of vertices of G.

Let us consider larger classes of graphs using context-free grammars. Given an alphabet N of *non-terminals* and an alphabet T of *terminals*, a *context-free grammar P* in Greibach normal form (without an axiom) may be seen as a finite subset of $N{\times}T{\times}N^*$: each production $p \longrightarrow aq$ with $a \in T$, $p \in N$ and $q \in N^*$, is replaced by a labelled transition $p \xrightarrow{a} q$. The set

$$P.N^* \ := \ \{\ pt \xrightarrow{a} ut \mid p \xrightarrow{a} u \ \in P \ \wedge \ t \in N^*\ \}$$

of the prefix transitions of P is called the *prefix transition graph* of P. For instance the prefix transition graph $P.\{x,y\}^*$ of this context-free grammar $P := \{x \xrightarrow{a} \epsilon\ ,\ x \xrightarrow{b} yx\}$ is the infinite replication of this following infinite connected component:



Recall that P is *reduced* if ϵ is a coroot of $P.N^*$, i.e. there is a path from every non-terminal (word) to ϵ.

In this context the deterministic context-free grammars are the *simple grammars*. Given an arbitrary reduced deterministic context-free grammar P, the bisimulation $\equiv_{P.N^*}$ of its prefix transitions corresponds to the equivalence of simple grammars, that can be solved polynomially according to the size of P and to its valuation [Ca 93]. Furthermore given an arbitrary deterministic grammar P (not necessary reduced), $\equiv_{P.N^*}$ corresponds to the decidable equality of infinitary simple languages [Ca 86].
For an arbitrary reduced grammar P (not necessary deterministic), $\equiv_{P.N^*}$ remains decidable [BBK 87]. Using the notion of self-proving relations [Co 83 b], we have shown that $\equiv_{P.N^*}$ is an equivalence induced by an effective idempotent morphism on N^*.

Proposition 3.1 [Ca 90 a] *Given any reduced context-free grammar P on a non-terminal alphabet N, we can construct an idempotent morphism h on N^* such that*

$$u \equiv_{P.N^*} v \quad \textit{iff} \quad h(u) = h(v) \qquad \textit{for every } u,v \in N^*.$$

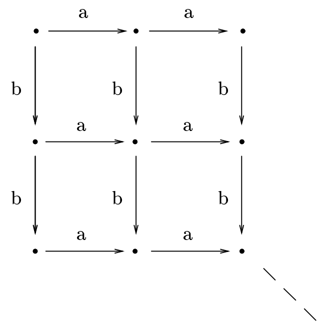
This construction has been improved to polynomial time [HJM 94] and allows to decide the equivalence of simple grammars polynomially according only to the size of the grammars.
From Proposition 3.1, it follows that $\equiv_{P.N^*}$ is effectively a *Thue congruence*: we can construct a finite binary relation R on N^* such that its generated congruence $\overset{*}{\underset{R}{\longleftrightarrow}}$ is equal to $\equiv_{P.N^*}$ [Ca 90 a]. Furthermore $\equiv_{P.N^*}$ is effectively a *rational transduction*: we can construct a transducer (finite graph labelled on $N^* \times N^*$ with an initial vertex and final vertices) recognizing $\equiv_{P.N^*}$. Finally this method has been extended to any context-free grammar.

Theorem 3.2 [CHS 92] *The greatest bisimulation on the prefix transition graph of any context-free grammar is decidable, is a Thue congruence and is a rational transduction.*

From [BCS 95] and for any context-free grammar P, we deduce that $\equiv_{P.N^*}$ is effectively a Thue congruence and is effectively a rational transduction.
We consider also the set

$$N^*.P.N^* \ := \ \{ \ spt \overset{a}{\longrightarrow} sut \mid p \overset{a}{\longrightarrow} u \ \in P \ \wedge \ s,t \in N^* \ \}$$

of all transitions of P, called the *transition graph* of P. When the associative concatenation over N is also commutative, the transition graph of any grammar P corresponds to its prefix transition graph, and is called the *commutative (prefix) transition graph* of P. For instance the commutative transition graph of $P = \{x \overset{a}{\longrightarrow} xx \ , \ x \overset{b}{\longrightarrow} xy\}$ is the following grid:



which is neither the prefix transition graph nor the transition graph of a grammar. However the transition graph of any grammar is reducible to its commutative transition graph: with every non-terminal word u we associate its equivalence class $[u]$ modulo the commutation relation $\overset{*}{\underset{R}{\longleftrightarrow}}$ for $R := \{(ab, ba) \mid a, b \in N\}$. So the bisimulation of grammar transition graphs is reducible to the bisimulation of grammar commutative transition graphs, which is decidable.

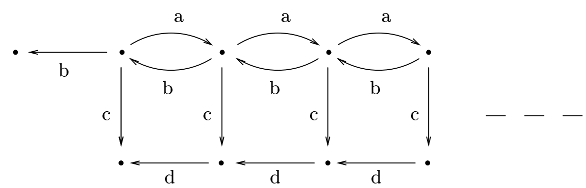
> **Theorem 3.3** **[CHM 93]** *The greatest bisimulation on the transition graph of any context-free grammar is an effective Thue congruence, hence is decidable.*

However the greatest bisimulation on the transition graph of any context-free grammar is not in general a rational transduction. For instance, the greatest bisimulation $\equiv_{N^*.P.N^*}$ of $P = \{x \overset{a}{\longrightarrow} \epsilon\ ,\ y \overset{b}{\longrightarrow} \epsilon\}$ on $N = \{x, y\}$ is equal to the non rational transduction $\{\ (u, v) \in N^* \times N^* \mid |u|_x = |v|_x \ \wedge\ |u|_y = |v|_y\ \}$ of the couple of words with the same occurrences of x and of y.
We will now consider the decidability of bisimulation for pushdown automata.

4 Bisimulation of pushdown automata

In this section we consider the bisimulation of pushdown automata, or only of real-time pushdown automata: the (strong) bisimulation does not take into account the empty transitions; the empty word is like an ordinary terminal. First we apply decision results on the equivalence of deterministic real-time pushdown automata to the bisimulation of their transition graphs (Theorems 4.2 and 4.3). Then we determine the structure of the transition graphs of pushdown automata : they are regular graphs meaning that they can be generated by deterministic graph grammars (Theorem 4.7). And we give two effective characterizations of regular graphs of finite degree (Theorems 4.8 and 4.9). Finally we show that the bisimulation is decidable on any graph which is regular by increasing path-length to a terminal coroot (Theorem 4.16).

As for context-free grammars over an alphabet N of non-terminals and an alphabet T of terminals, a *real-time pushdown automaton* P (without axiom and final states) may be seen as a finite subset of $Q.N \times T \times Q.N^*$ where Q is an alphabet of *states* disjoint of N. So the transition graph $N^*.P.N^*$ of P is the infinite replication of its prefix transition graph $P.N^*$ called its *pushdown transition graph*. For instance the pushdown transition graph of the (real-time pushdown) automaton $\{px \xrightarrow{a} pxx \ , \ px \xrightarrow{b} p \ , \ px \xrightarrow{c} q \ , \ qx \xrightarrow{d} q\}$ where $N = \{x\}$, $Q = \{p,q\}$, $T = \{a,b,c,d\}$ is the following graph:



Taking a new letter q, every grammar P is transformed into the following automaton $\{ \ qp \xrightarrow{a} qu \mid p \xrightarrow{a} u \ \in P \ \}$ with a single state $Q = \{q\}$ and such that its pushdown transition graph is isomorphic to the prefix transition graph of P. The converse is false: the previous graph is not a connected component of the prefix transition graph of any grammar [Ca 90 b]. Thus pushdown transition graphs are more general than grammar transition graphs. Including Q in N, a real-time pushdown automaton P is a *labelled word rewriting system*, i.e. a finite subset of $N^* {\times} T {\times} N^*$. Note that the prefix transition graph of the following rewriting system $\{x \xrightarrow{a} \epsilon \ , \ x^2 \xrightarrow{b} x^3\}$ is not a pushdown transition graph. However the prefix transition graphs of labelled word rewriting systems have effectively the same accessible subgraphs than the pushdown transition graphs [Ca 90 b]. This correspondence remains true for the connected components.

Proposition 4.1 *The prefix transition graphs of labelled word rewriting systems and the pushdown transition graphs have effectively the same connected components and the same accessible subgraphs.*

We now consider the decidability of bisimulation on pushdown transition graphs. Given any graph G in $V {\times} T {\times} V$, we denote by $L(G,r,E)$ the set of path labels of G from $r \in V$ to any vertex in $E \subseteq V$. In particular $L(P.N^*,r,F.N^*)$ is the *language accepted* by P from *axiom* $r \in Q.N^*$ with *acceptance by final states* in $F \subseteq Q$.
Recall that P is *reduced* if from every word in $Q.N$ (or in $Q.N^*$), there is a path ending to some state; for instance the previous automaton is reduced. For an arbitrary reduced real-time *dpda* (*d*eterministic *p*ush*d*own *a*utomaton) P, $\equiv_{P.N^*}$ is inter-reducible to its decidable equivalence problem with acceptance by empty stack [OIH 80]:

$$u \equiv_{P.N^*} v \quad \text{iff} \quad L(P.N^*,u,Q) = L(P.N^*,v,Q).$$

Theorem 4.2 [OIH 80] *The greatest bisimulation on any reduced and deterministic pushdown transition graph is effectively a rational transduction, hence is decidable.*

For an arbitrary real-time dpda P (not necessary reduced), $\equiv_{P.N^*}$ is inter-reducible to its equivalence problem with acceptance on any state:

$$u \equiv_{P.N^*} v \quad \text{iff} \quad L(P.N^*,u,Q.N^*) = L(P.N^*,v,Q.N^*).$$

This problem is decidable even for any subset of final states [Ro 86]. In fact for pushdown automata (respectively real-time/deterministic) the equivalence problem with acceptance on final states is inter-reducible to the equivalence problem with acceptance on any state.

Theorem 4.3 [Ro 86] *The bisimulation on any deterministic pushdown transition graph is decidable.*

Nevertheless for an arbitrary non reduced real-time dpda P, we do not know if $\equiv_{P.N^*}$ is a rational transduction, but in this case we cannot construct a rational transducer, by reducing the undecidable Post's Correspondence Problem [Co 81].

A way to consider bisimulation on pushdown transition graphs is to determine their structures. The structure of pushdown transition graphs has been originally studied in [MS 85]. Let us give another but effective characterization of these graphs by using graph grammars. We take a set of labels with non null arities. A *hyperarc* is a word $as_1 \ldots s_p$ labelled by a of arity $p > 0$ and joining in order the vertices $s_1, \ldots, s_p$. In particular an arc $s \xrightarrow{a} t$ is the word ast with a of arity 2. A *hypergraph* H is a set of hyperarcs. A *graph grammar* R is a finite set of rules of the form $ax_1 \ldots x_p \longrightarrow H$ where $x_1, \ldots, x_p$ are distinct vertices and H is a finite hypergraph. The labels of the left hand sides of R are the *non-terminals* of R. The other labels in R are the *terminals* and are of arity 2 (they label arcs only). We say that R is *deterministic* if there is only one rule by non-terminal. A *rewriting* $M \underset{R}{\longrightarrow} N$ consists in choosing a non-terminal hyperarc $X = as_1 \ldots s_p$ in M and a rule $ax_1 \ldots x_p \longrightarrow H$ in R to be applied; the vertices x_i in H indicate how to replace X by H:

$$N \;=\; (M - X) \;\cup\; \{\; bg(t_1) \ldots g(t_q) \mid bt_1 \ldots t_q \in H\}$$

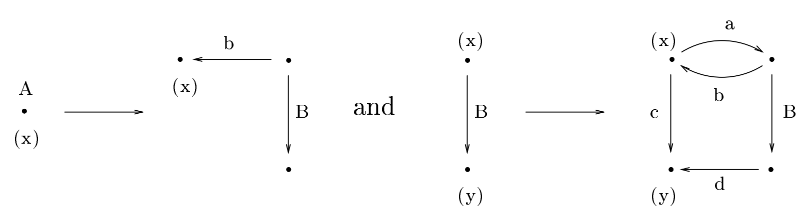
for some matching function g mapping x_i to s_i, and the other vertices of H injectively to vertices outside M; this rewriting is also denoted by $M \underset{R,X}{\longrightarrow} N$. Note that $\underset{R}{\longrightarrow}$ is not in general a functional relation, even when R is deterministic. Nevertheless,

$$M \underset{R,X_1}{\longrightarrow} \circ \ldots \circ \underset{R,X_n}{\longrightarrow} N \quad \text{iff} \quad M \underset{R,X_{\pi(1)}}{\longrightarrow} \circ \ldots \circ \underset{R,X_{\pi(n)}}{\longrightarrow} N$$

for any $X_i \in M$ and for any permutation π on {1,...,n}. Thus, it makes sense to define a *complete parallel rewriting* $\underset{R}{\Longrightarrow}$ as follows :

$$M \underset{R}{\Longrightarrow} N \quad \text{if} \quad M \underset{R,X_1}{\longrightarrow} \circ \ldots \circ \underset{R,X_n}{\longrightarrow} N$$

where $X_1, \ldots, X_n$ are all the non-terminal hyperarcs of M. We denote by $[M] = \{\; ast \in M \mid a$ is a terminal of $R\;\}$ the set of terminal arcs of M. A graph G is *generated by a deterministic graph grammar* R from a hypergraph H if G is isomorphic to $\bigcup_{n \geq 0}[H_n]$ where $H_0 \;=\; H$ and $H_n \underset{R}{\Longrightarrow} H_{n+1}$ for every $n \geq 0$. For instance the following deterministic graph grammar :



generates from the hypergraph $\{Ax\}$ the previous pushdown transition graph.

Definition 4.4 A *regular graph* is a graph generated by a deterministic graph grammar from a finite hypergraph.

These graphs are the equational graphs of [Co 90]. Restricted to trees, a regular tree is a tree with a finite number of nonisomorphic subtrees, i.e. the unfolded tree of a finite graph. In particular the first example of the previous section gives a regular graph with a non regular unfolded tree. Let us give basic properties of regular graphs.

Proposition 4.5 *We have the following properties:*
a) *Every accessible subgraph of a regular graph is effectively a regular graph;*
b) *Every connected component of a regular graph is effectively a regular graph;*
c) *Every regular graph has effectively a finite number of nonisomorphic connected components.*

Let us give some correspondences between regular graphs and prefix transition graphs of word rewriting systems. The set of vertices of any prefix transition graph $R.N^*$ is equal to $(Dom(R) \cup Im(R))N^*$, hence is rational. We also have an effective rational language for the vertices accessible from a given axiom [Bü 64], or better for the vertices accessible from any rational set of vertices.

Proposition 4.6 *The prefix derivation of any (unlabelled) word rewriting system is effectively a rational transduction.*

This entails that any pushdown transition graph is a regular graph [Ca 90 b] of *finite degree*, i.e. every vertex belongs only to a finite number of arcs. The converse is false because every pushdown transition graph is either empty or infinite. This converse becomes true if we restrict pushdown transition graphs to their accessible subgraphs, called *accessible pushdown transition graphs*.

Theorem 4.7 [Ca 90 b] *We have the following properties:*
a) *The accessible pushdown transition graphs are effectively the rooted components of pushdown transition graphs, and are effectively the rooted regular graphs of finite degree;*
b) *The connected components of pushdown transition graphs are effectively the connected regular graphs of finite degree.*

For instance every deterministic graph grammar generating a rooted graph G of finite degree, is mapped effectively into a pushdown automaton with an axiom such that its accessible transition graph is isomorphic to G, and the reverse transformation is also effective.

One way to obtain all regular graphs of finite degree is to take the restrictions of pushdown transition graphs $P.N^*$ to any rational subset $V \subseteq Q.N^*$ of configurations :

$$(P.N^*)_{|V} \;=\; \{\, p \xrightarrow{a} q \;\in P.N^* \mid p, q \in V \,\} \;.$$

Theorem 4.8 *The following families of graphs coincide effectively:*
a) *The regular graphs of finite degree;*
b) *The rational restrictions of prefix transition graphs of labelled rewriting systems;*
c) *The rational restrictions of pushdown transition graphs.*

Another way to obtain all regular graphs of finite degree is to apply a rule by prefix and only if its right context belongs to a given rational language. A similar method has been defined in [Ch 82]. More precisely, a *rationally controlled labelled rewriting system* (R, f) is a labelled rewriting system R with a mapping f from R into the rational sets of non-terminal words. The prefix transition graph of (R, f) is the set

$$R.f \;:=\; \{\, uw \xrightarrow{a} vw \mid u \xrightarrow{a} v \;\in R \,\wedge\, w \;\in f(u \xrightarrow{a} v) \,\}$$

of prefix transitions of R whose right contexts are allowed by f.

Theorem 4.9 *The following families of graphs coincide effectively:*
a) *The regular graphs of finite degree;*
b) *The prefix transition graphs of rationally controlled labelled rewriting systems;*
c) *The (pushdown) transition graphs of rationally controlled pushdown automata.*

Let us give some basic closure properties of rationally controlled prefix transition graphs. In particular and despite the class of rationally controlled prefix transition graph is not closed by quotient by the greatest bisimulation, it is closed by any morphism on vertices.

Proposition 4.10 *The class of rationally controlled prefix transition graphs is closed effectively by union, difference, intersection, morphism, and rational restriction.*

Note that the class of rationally controlled prefix transition graphs is not closed by inverse morphism. Let us apply Proposition 4.10 to Proposition 3.1.

Corollary 4.11 [Ca 90 a] *Any couple (P, N) of a reduced context-free grammar P on a non-terminal alphabet N can be transformed into another couple $(\overline{P}, \overline{N})$ such that $\overline{P}.\overline{N}^*$ is the quotient of $P.N^*$ by its greatest bisimulation.*

It follows that the unfolded tree from the root of the previous regular graph is not the unfolded tree of any (reduced) cf-grammar.

Given a deterministic graph grammar S generating a (regular) graph G of finite degree, we give a general method to transform S into another deterministic graph grammar generating G. Let G be any labelled graph and let g be a *graduation* of G, i.e. a mapping from the vertex set V_G of G into the integer set $\mathbb{N}$. For every $n \geq 0$, we denote by

$$(G)_{g,n} \ := \ \{ \ s \xrightarrow{a} t \ \in G \mid g(s) \leq n \ \wedge \ g(t) \leq n \ \wedge \ (g(s) < n \ \vee \ g(t) < n) \ \}$$

the first n-th slides of the *decomposition* of G by g, and we denote by

$$[G]_{g,n} \ := \ \{ \ s \in V_G \mid g(s) = n \ \vee \ (s \in V_{G-(G)_{g,n}} \ \wedge \ g(s) < n) \ \}$$

the n-th *frontier* of G by g.
We say that G is a *regular graph according to* g if there exist a deterministic graph grammar S and a hyperarc X such that for every $n \geq 0$, S generates from X by n parallel rewritings the graph $(G)_{g,n}$ of terminal arcs with the vertex set $[G]_{g,n}$ of non-terminal hyperarcs, i.e.

$$\forall \ n \geq 0 \ \ \exists \ H, \ \ X \underset{S}{\Longrightarrow}^{n} H \ \ \wedge \ \ [H] = (G)_{g,n} \ \ \wedge \ \ V_{H-[H]} = [G]_{g,n} \ ;$$

except when $[G]_{g,n} = \emptyset$ and $(G)_{g,n} \neq G$, $V_{H-[H]}$ is restricted to a vertex not in V_G.
Given any vertices s, t of G, we denote by

$$d_G(s,t) \ := \ min(\{ \ |u| \mid u \in L(G \cup G^{-1}, s, t) \ \} \ \cup \ \{\infty\})$$

the *distance* in G between s and t.
Furthermore we denote by $((G))_{g,s}$ the connected component of $G - (G)_{g,g(s)}$ containing s (may be empty), and by $[[G]]_{g,s} \ := \ [G]_{g,g(s)} \cap V_{((G))_{g,s}}$ its frontier by g.
We write $s \sim_h t$ if h is an isomorphism from $((G))_{g,s}$ to $((G))_{g,t}$ such that $h([[G]]_{g,s}) = [[G]]_{g,t}$ and $h(s) = t$. We say that g is *vertex independent* if the decomposition is identical for isomorphic connected components with the same frontier:

$$s \sim_h t \quad \Longrightarrow \quad g(r) - g(s) = g(h(r)) - g(t) \quad \text{for every vertex } r \text{ of } ((G))_{g,s}.$$

Note that if G is a regular graph according to g then g is *of bounded connection* in the following sense :

$$\exists \ b \geq 0 \ \ \forall \ n \ \geq 0 \ \ \forall \ s,t \in [G]_{g,n} \ , \ \ d_{G-(G)_{g,n}}(s,t) = \infty \ \ \vee \ \ d_{G-(G)_{g,n}}(s,t) \leq b.$$

The converse is true for any connected regular graph of finite degree and with a vertex independent graduation.

Lemma 4.12 *Given a connected rationally controlled rewriting system (R,f) and a vertex independent graduation $g : V_{R.f} \longrightarrow \mathbb{N}$, the following two properties are equivalent :*
a) *$R.f$ is a [resp. is an effective] regular graph according to g,*
b) *g is of bounded connection [resp. g^{-1} is recursive].*

Note that if G is a regular graph according to g then $g(G)$ is again a regular graph. The converse is false except for instance when g is a ϵ-free morphism on a rationally controlled prefix transition graph.

Proposition 4.13 *Any rationally controlled prefix transition graph is an effective regular graph according to any ϵ-free morphism.*

Thus (and as already seen in proof $(b) \Rightarrow (a)$ of Theorem 4.9), $R.f$ can be generated by vertices of increasing length with a deterministic graph grammar. Another consequence of Lemma 4.12 is that any connected graph $R.f$ can be generated with a deterministic graph grammar by increasing distance $d(u,E) \ := \ min\{ \ d(u,v) \mid v \in E \ \}$ of any vertex u from a given nonempty finite set E of vertices.

Proposition 4.14 [Ca 90 b] *Any connected rationally controlled prefix transition graph is an effective regular graph according to the distance from any nonempty finite set of vertices.*

We are ready to extend the method defined in [BBK 87] to decide the bisimulation for a general subclass of regular graphs including the prefix transition graphs of reduced context-free grammars.
Note that the prefix transition graph of any reduced context-free grammar has ϵ as terminal coroot. Recall that a vertex c of a graph G is *terminal* if it is source of no arc :

$$\forall \, a \, \forall \, t \ \ (c \xrightarrow{a} t) \ \notin G \ ,$$

and that c is a *coroot* if there is a path from any vertex to c, i.e.

$$\forall \, s \in V_G \ \ s \underset{G}{\Longrightarrow}^* c \ .$$

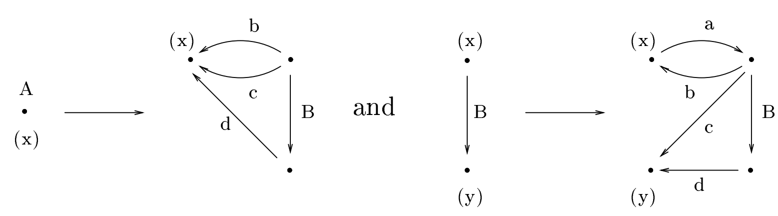
In particular, a terminal coroot of a graph is unique. Given any graph G with a terminal coroot c, we define the *valuation* $\| \, s \, \|$ of any vertex s as being the minimal length of the pathes from s to c, i.e.

$$\| \, s \, \| \ := \ min\{ \ |w| \mid w \in L(G,s,c) \ \}.$$

For any reduced context-free grammar on a non-terminal alphabet N, the valuation on N^* is an ϵ-free morphism, hence Proposition 4.13 can be applied.

Corollary 4.15 *The prefix transition graph of any reduced context-free grammar is an effective regular graph according to its valuation.*

The converse is false : there exist graphs regular by valuation which are not isomorphic to any restriction of the prefix transition graph of any reduced context-free grammar. For instance and by identifying the two terminal vertices of the previous graph, we obtain a graph with a terminal coroot which cannot be obtained with a reduced context-free grammar [CM 90], and that can be generated by increasing valuation according to the following deterministic graph grammar :



A basic property is that for any graph with a terminal coroot, two bisimilar vertices have the same valuation. In a way similar to [BBK 87], we reduce the decidability of the bisimulation on any regular graph according to its valuation, to the decidability on the finite subgraph of all vertices having a valuation bounded by a computable value.

Theorem 4.16 *The bisimulation on any regular graph according to its valuation is decidable.*

Contrary to Corollary 4.11, the class of graphs regular by valuation is not closed by quotient by the greatest bisimulation. But for any finite degree graph regular by valuation, if its quotient by a bisimulation is regular then this quotient is regular by valuation.

5 Bisimulation of term context-free grammars

In this section, we consider the bisimulation of context-free grammars on terms (Definition 5.1). We define two subclasses of context-free grammars such that their accessible prefix transition graphs are exactly and effectively the rooted regular graphs of finite out-degree (Theorem 5.3), resp. of finite degree (Theorem 5.4). On the other hand, we show that the prefix transition graphs of context-free grammars and the regular graphs of finite out-degree have the same unfolded trees (Proposition 5.6), hence the bisimulation on these graphs has the same decision problem. Finally, the bisimulation decidability of deterministic context-free grammars is inter-reducible to the well-known equivalence problem of dpda (Proposition 5.9).

We introduce some definitions and usual notations. The set $T(F)$ of (finite) *terms* on a graded alphabet $F = \bigcup_{n\geq 0} F_n$ is the least language in F^+ such that $ft_1 \ldots t_n \in T(F)$ for every $f \in F_n$ (of arity $n \geq 0$) and every $t_1, \ldots, t_n \in T(F)$. A *constant* is an element of F_0. A *subterm* s of a term t is a term which is a factor of t ($usv = t$ for some u, v). The set $F(t)$ of symbols of some term $t = ft_1 \ldots t_n$ on F is defined recursively on its structure by $F(t) \ := \ \{f\} \ \cup \ \bigcup_{1\leq i\leq n} \ F(t_i)$. We denote by $E(t) \ := \ F(t) \cap E$ the set of symbols in $E \subseteq F$ of a term t. Furthermore the *height* $h(t)$ of any term $t = ft_1 \ldots t_n$ is the greatest length of its branches : $h(t) \ := \ max(\{\ 1 + h(t_i) \mid 1 \leq i \leq n\} \cup \{0\})$.
We take a denumerable set $V = \{x_1, x_2, \ldots\}$ of *term variables* (symbols of arity zero), disjoint from F. The context-free grammars on words are extended to terms. Recall that T is the set of terminals and F will be the (graded) set of non-terminals.

Definition 5.1 A *term context-free grammar* on $F \cup V$ is a finite set of labelled rules $fx_1 \ldots x_n \stackrel{a}{\longrightarrow} t$ where $n \geq 0$, $f \in F_n$, $a \in T$ and $t \in T(F \cup \{x_1, \ldots, x_n\})$.

In particular, a term context-free grammar is a *labelled term rewriting system*, i.e. a finite set of labelled rules $s \stackrel{a}{\longrightarrow} t$ of terms s, t on $F \cup V$ such that $V(t) \subseteq V(s)$. Recall that a *substitution* σ is a mapping from V into $T(F \cup V)$ which is extended into a morphism on $T(F \cup V)$: $\sigma(ft_1 \ldots t_n) \ = \ f\sigma(t_1) \ldots \sigma(t_n)$. We denote by $t[t_1, \ldots, t_n]$ the substitution $\sigma(t)$ of t defined by $\sigma(x_1) = t_1, \ldots, \sigma(x_n) = t_n$ and $\sigma(x) = x$ for every other variable x. The prefix transition graph of any context-free grammar on words can be extended to any context-free grammar P on terms: the set

$$P.T(F)^* \ := \ \{\ ft_1 \ldots t_n \stackrel{a}{\longrightarrow} t[t_1, \ldots, t_n] \mid fx_1 \ldots x_n \stackrel{a}{\longrightarrow} t \ \in P \ \wedge \ t_1, \ldots, t_n \in T(F)\ \}$$

of the prefix transitions of P is the *prefix transition graph* of P; its accessible subgraphs are the *accessible prefix transition graphs* of P. More generally, the prefix transition graph of any term rewriting system P is the set

$$P.T(F)^* \;:=\; \{\; \sigma(s) \xrightarrow{a} \sigma(t) \mid s \xrightarrow{a} t \;\in P \;\wedge\; \sigma \text{ is a substitution } \}$$

of its prefix transitions. Note that any prefix transition graph is of *finite out-degree* : every vertex is source of only a finite number of arcs.

For instance, taking $F = \{f, g, h\}$ with f, g, h of respective arities $2, 1, 0$, the accessible prefix transition graph $P.T(F)^*/fhh$ of this (linear) context-free grammar

$$P := \{fx_1x_2 \xrightarrow{a} fgx_1x_2 \;,\; fx_1x_2 \xrightarrow{b} fx_1gx_2\}$$

and from axiom fhh is the previous grid, which is not a regular graph. In fact, the term fgx_1x_2 is not entire in the following sense : a term t is *entire* if for every subterm s of t, either s has the same variables as t, or s is a variable or s has no variable; for instance, $fx_1fx_2x_1$ and $fgcx_1$ are entire terms. By extension, a rewriting system is entire if for every rule $s \xrightarrow{a} t$, the terms s and t are entire.

Accessible prefix transition graphs of term cf-grammars generalize rooted regular graphs of finite out-degree. More precisely, it is known that the prefix transition graphs of entire systems, or of cf-grammars, and the regular graphs of finite out-degree have the same accessible subgraphs [Ca 92]. Let us refine this characterization by restricting as much as possible the class of entire cf-grammars.

Definition 5.2 A term cf-grammar P is *standard* if every rule $s \xrightarrow{a} t$ of P satisfies the following three properties :

(i) $h(t) \leq 2$

(ii) $t \notin V \implies V(t) = V(s)$

(iii) $s \notin F_0 \implies F_0(t) = \emptyset$.

For instance with symbols f, g, h of respective arities $2, 1, 0$, these entire cf-grammars $\{gx \xrightarrow{a} gggx\}$, $\{fxy \xrightarrow{a} gx\}$, $\{gx \xrightarrow{a} fhx\}$ do not satisfy respectively (i),(ii),(iii) of Definition 5.2. This definition allows to refine the class of entire cf-grammars in such a way that their accessible prefix transition graphs are the rooted regular graphs of finite out-degree.

Theorem 5.3 *The following families of graphs coincide effectively:*

a) *The rooted regular graphs of finite out-degree;*

b) *The accessible prefix transition graphs of entire rewriting systems;*

c) *The accessible prefix transition graphs of standard and entire cf-grammars.*

This correspondence becomes false for the connected components: it suffices to take the following standard and entire cf-grammar $P := \{fx_1x_2 \xrightarrow{a} x_1 \;,\; fx_1x_2 \xrightarrow{b} x_2\}$ on $F = \{f, h\}$ with f, h of respective arities $2, 0$, then its prefix transition graph $P.T(F)^*$ has the inverse grid as subgraph, so has an undecidable monadic theory [Se 91], hence is not regular, because every regular graph has a decidable monadic theory [Co 90].

Let us restrict Theorem 5.3 to characterize the rooted regular graphs of finite degree. We say that a term $t = ft_1 \ldots t_n$ is *perfect* if for every $1 \leq i \leq n$, $h(t_i) = h(t) - 1$ and by induction t_i is perfect. For instance, with functions f, g, h of respective arities $2, 1, 0$, the term $fghfhh$ is perfect but the term $gfhgh$ is not perfect. By extension, a rewriting system is perfect if for every rule $s \xrightarrow{a} t$, the terms s and t are perfect. In particular, the left hand sides of any cf-grammar are perfect. Furthermore, we say that a term rewriting system P is *constant-separated* if it satisfies the following weaker condition than condition (iii) of Definition 5.2 :

$$V(t) \neq \emptyset \implies F_0(t) = \emptyset \qquad \text{for any rule } s \xrightarrow{a} t \text{ of } P.$$

In particular, every standard cf-grammar is constant-separated. The restriction of Theorem 5.3 to perfect and constant-separated systems, gives the regular graphs of finite degree.

Theorem 5.4 *The following families of graphs coincide effectively:*
a) *The rooted regular graphs of finite degree;*
b) *The accessible prefix transition graphs of perfect, constant-separated and entire rewriting systems;*
c) *The accessible prefix transition graphs of perfect, standard and entire cf-grammars.*

Now we consider the trees obtained by prefix unfolding of term context-free grammars.

Definition 5.5 A *context-free tree* is an unfolded tree of the prefix transition graph of a context-free grammar on terms.

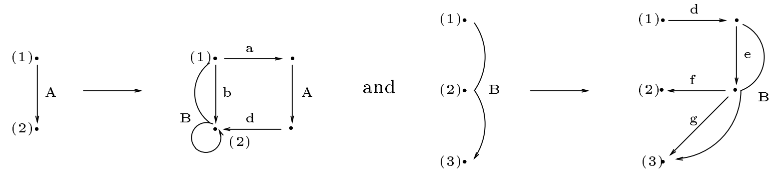
Although the accessible prefix transition graphs of term context-free grammars is a proper extension of the rooted regular graphs of finite out-degree, their unfolded trees are the same.

Proposition 5.6 *The context-free trees coincide effectively with the unfolded trees of finite out-degree regular graphs.*

Thus the equality problem of context-free trees is the same that the decidability of bisimulation on regular graphs of finite out-degree.

Corollary 5.7 *The bisimulation on prefix transition graphs of term cf-grammars is inter-reducible to the bisimulation on regular graphs of finite out-degree.*

In fact, the unfolded trees of regular graphs of finite out-degree is a proper extension of the unfolded trees of regular graphs of finite degree. For instance, the unfolded tree from the root of the regular graph of finite out-degree and generated by the following deterministic graph grammar :



is not the unfolded tree of any regular graph of finite degree. The unfolded trees of cf-grammars on words and on terms, of pushdown automata, and of regular graphs can be classified.

Proposition 5.8 *We have the following hierarchy on trees :*

	finite trees
$\subset$	*regular trees of finite degree*
$=$	*unfolded trees of finite graphs*
$\subset$	*unfolded trees of context-free grammars on words*
$\subset$	*unfolded trees of pushdown automata*
$=$	*unfolded trees of regular graphs of finite degree*
$\subset$	*unfolded trees of regular graphs of finite out-degree*
$=$	*unfolded trees of context-free grammars on terms*
$\subset$	*unfolded trees of regular graphs.*

This hierarchy is valid for deterministic trees, except that the last inclusion is an equality.

As for the bisimulation on regular graphs of finite degree, we distinguish the deterministic case for the bisimulation on term cf-grammars. Let us recall the equivalence problem for dpda. A *dpda* P (not necessary real-time) is a finite subset of $Q.N \times T \cup \{\epsilon\} \times Q.N^*$ satisfying the following two conditions:

(i) P is deterministic, i.e. if $s \xrightarrow{a} u$, $s \xrightarrow{a} v \in P$ then $u = v$ (even if $a = \epsilon$)

(ii) if $s \xrightarrow{\epsilon} u$, $s \xrightarrow{a} v \in P$ then $a = \epsilon$.

The *equivalence problem for dpda* is to decide whether

$$L(P.N^*, u, F.N^*) \;=\; L(P.N^*, v, F.N^*)$$

for any dpda P, any subset F of states and any configurations $u, v \in Q.N^*$. Note that the acceptance is by final states but the problem is not more difficult if the acceptance is by empty stack, or also is on all states. Thus by identifying source and target of ϵ-transitions, this problem is reducible to the bisimulation of deterministic regular graphs; and the inverse transformation also holds.

Proposition 5.9 *The following problems are inter-reducible :*

a) *The equivalence for dpda*

b) *The bisimulation on deterministic regular graphs*

c) *The equality of deterministic context-free trees.*

Despite the last approach of [Me 92], the equivalence problem for dpda remains open.

In fact, the deterministic context-free trees correspond to the algebraic terms (trees)

[Co 83 a]. Recall that a recursive program *scheme* [Ni 75], [Gu 81] P on $F \cup V$ is an unlabelled term cf-grammar, i.e. a finite set of rules

$$fx_1 \ldots x_n \longrightarrow t \quad \text{with} \quad V(t) \subseteq \{x_1, \ldots, x_n\}$$

such that P is *functional* : $\quad fx_1 \ldots x_n \longrightarrow s \;\wedge\; fx_1 \ldots x_n \longrightarrow t \quad \Longrightarrow \quad s = t$

and P is in *Greibach form* : $\quad fx_1 \ldots x_n \longrightarrow t \quad \Longrightarrow \quad t(1) \in F - N_P$

where $N_P := \{ f \in F_n \mid fx_1 \ldots x_n \in Dom(P) \}$ is the set of *non-terminals* of P.
An *algebraic term* is an unfolded (finite or infinite) term $P^\omega(t)$ on $F - N_P$ obtained by a scheme P from a term $t = ft_1 \ldots t_n \in T(F)$, as follows :

$$P^\omega(ft_1 \ldots t_n) := \begin{cases} fP^\omega(t_1) \ldots P^\omega(t_n) & \text{if } f \notin N_P \\ P^\omega(s[t_1, \ldots, t_n]) & \text{if } fx_1 \ldots x_n \longrightarrow s \in P \,. \end{cases}$$

By a simple replacement of vertex labels by arc labels, the algebraic terms correspond to the deterministic context-free trees.

Lemma 5.10 *The equality problem of deterministic context-free trees is inter-reducible to the equality problem of algebraic terms.*

Proposition 5.9 with Lemma 5.10 yield a known result.

Corollary 5.11 [CG 87] *The equivalence problem for dpda is inter-reducible to the equality problem of algebraic terms.*

6 Conclusion

We have shown (Theorem 4.16) that the bisimulation is decidable for any reduced pushdown automata having a regular transition graph according to the path-length to its terminal vertices. However the decidability of the bisimulation for any pushdown automaton is an open problem.

Problem 6.1 *Is bisimulation decidable for pushdown automata ?*

From Theorem 4.7 (a), problem 6.1 is inter-reducible to the decidability of bisimulation of finite degree regular graphs. So Problem 6.1 is a restricted case of the following problem.

Problem 6.2 *Is bisimulation decidable for regular graphs of finite out-degree ?*

But this problem restricted to the deterministic case, is inter-reducible to the still opened famous equivalence problem of deterministic pushdown automata (Theorem 5.9). Finally, Problem 6.2 is a restricted case of the following problem.

Problem 6.3 *Is bisimulation decidable for regular graphs ?*

References

[BBK 87] J. Baeten, J. Bergstra and J. Klop *Decidability of bisimulation equivalence for processes generating context-free languages*, PARLE 87, LNCS 259, pp. 94–111, 1987. Extended version in Journal of the ACM 40-3, pp. 653–682, 1993.

[BCS 95] O. Burkart, D. Caucal and B. Steffen *An elementary bisimulation decision procedure for arbitrary context-free processes*, MFCS 95, LNCS 969, pp. 423–433, 1995.

[BN 84] L. Boasson and M. Nivat *Centers of context-free languages*, Internal Report LITP 84–44, 1984.

[Bü 64] R. Büchi *Regular canonical systems*, Archiv für Mathematische Logik und Grundlagenforschung 6, pp. 91–111, 1964
or in *The collected works of J. Richard Büchi*, edited by S. Mac Lane and D. Siefkes, Springer-Verlag, New York, pp. 317–337, 1990.

[Ca 86] D. Caucal *The equality of infinitary simple languages is decidable*, STACS 86, LNCS 210, pp. 37–48, 1986.

[Ca 90 a] D. Caucal *Graphes canoniques de graphes algébriques*, RAIRO-TIA 24-4, pp. 339–352, 1990.

[Ca 90 b] D. Caucal *On the regular structure of prefix rewriting*, CAAP 90, LNCS 431, pp. 87–102, 1990, extended in TCS 106, pp. 61–86, 1992.

[Ca 92] D. Caucal *Monadic theory of term rewritings*, 7^{th} IEEE Symp., LICS 92, pp. 266–273, 1992.

[Ca 93] D. Caucal *A fast algorithm to decide on the equivalence of stateless dpda*, RAIRO-TIA 27-1, pp. 23–48, 1993.

[CM 90] D. Caucal and R. Monfort *On the transition graphs of automata and grammars*, WG 90, LNCS 484, pp. 311–337, 1990.

[Ch 82] L. Chottin *Langages algébriques et systèmes de réécriture rationnels*, RAIRO-TIA 16-2, pp. 93–112, 1982.

[CHM 93] S. Christensen, Y. Hirshfeld and F. Moller *Bisimulation equivalence is decidable for basic parallel processes*, CONCUR 93, LNCS 715, pp. 143–157, 1993.

[CHS 92] S. Christensen, H. Hüttel and C. Stirling *Bisimulation equivalence is decidable for all context-free processes*, CONCUR 92, LNCS 630, pp. 138–147, 1992, appeared in Information and Computation 121-2, pp. 143–148, 1995.

[Co 81] B. Courcelle *The simultaneous accessibility of two configurations of two equivalent dpda's*, Information Processing Letter 12-3, pp. 111–114, 1981.

[Co 83 a] B. Courcelle *Fundamental properties of infinite trees*, TCS 25, pp. 95–169, 1983.

[Co 83 b] B. Courcelle *An axiomatic approach to the* KH *algorithms*, Mathematical System Theory 16, pp. 191–231, 1983.

[Co 90] B. Courcelle *Graph rewriting: an algebraic and logic approach*, Handbook of TCS, Vol. B, Elsevier, pp. 193–242, 1990.

[CG 87] B. Courcelle and J. Gallier *Decidable subcases of the equivalence problem for recursive program schemes*, RAIRO-TIA 21-3, pp. 245–286, 1987.

[Gu 81] I. Guessarian *Algebraic Semantics*, LNCS 99, 1981.

[HJM 94] Y. Hirshfeld, M. Jerrum and F. Moller *A polynomial algorithm for deciding bisimilarity of normed context-free processes*, Dagstuhl, Algorithms in Automata Theory, 1994, will appear in TCS.

[Me 92] V. Meitus *Decidability of the equivalence problem for deterministic pushdown automata*, Kibernetika 5, pp. 20–45, 1992.

[MS 85] D. Muller and P. Schupp *The theory of ends, pushdown automata, and second-order logic*, TCS 37, pp. 51–75, 1985.

[Ni 75] M. Nivat *On the interpretation of polyadic recursive schemes*, Symposia Mathematica 15, Academic Press, 1975.

[OIH 80] M. Oyamaguchi, Y. Inagaki and N. Honda *The equivalence problem for realtime strict deterministic languages*, Information and Control 45, pp. 90–115, 1980.

[PT 87] R. Paige and R. Tarjan *Three partition refinement algorithms*, SIAM J. COMPUT. 16-6, pp. 973–989, 1987.

[Pa 81] D. Park *Concurrency and automata on infinite sequences*, LNCS 104, pp. 167–183, 1981.

[Ro 86] V. Romanovskii *The equivalence problem for real-time deterministic pushdown automata*, Kibernetika 2, pp. 12–23, 1986.

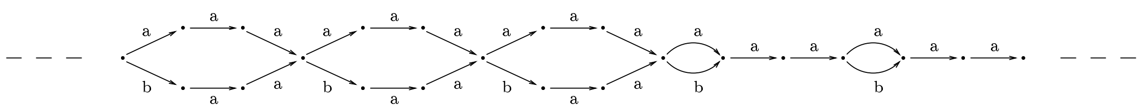
[Se 91] D. Seese *The structure of the models of decidable monadic theories of graphs*, Annals of Pure and Applied Logic 53, pp. 169–195, 1991.

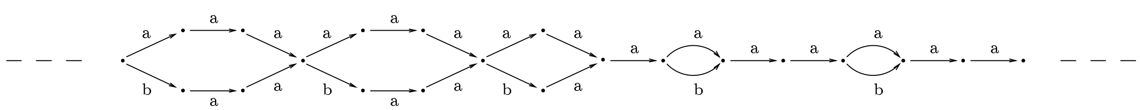
A Appendix

We give here all the proofs of new results.

A.a Bisimulation

We need some basic facts on reductions. First we give below two bi-infinite connected graphs G and H such that G and H are interreducible: $G \longrightarrow H$ and $H \longrightarrow G$, but G and H are not isomorphic.





Given a bisimulation R on G which is an equivalence, usually called a *congruence*, we define the *quotient* of G by R as follows:

$$G/R \ := \ \{ \ [p]_{_R} \overset{a}{\longrightarrow} [q]_{_R} \mid p \overset{a}{\longrightarrow} q \in G \ \}$$

where $[p]_{_R} \ := \ \{q \mid p \ R \ q\}$ is the equivalence class of p according to R. The mapping $[\ ]_{_R}$ is a bisimulation from G to G/R.

Lemma A.1 *Given a congruence R of a graph G, $[\ ]_{_R} : G \longrightarrow G/R$ is a reduction.*

Proof.
It suffices to prove that $[\ ]_{_R}$ is a bisimulation.
Let $p \overset{a}{\longrightarrow} q$. By definition of G/R, $[p]_{_R} \overset{a}{\longrightarrow} [q]_{_R}$ hence $[\ ]_{_R}$ is a simulation.
Conversely if $[p]_{_R} \overset{a}{\longrightarrow} [q]_{_R}$ then there is $p' \overset{a}{\longrightarrow} q'$ with $p' \ R \ p$ and $q' \ R \ q$.
As R is a simulation there is q'' such that $p \overset{a}{\longrightarrow} q''$ and $q' \ R \ q''$. So $[q'']_{_R} = [q]_{_R}$.
$\square$

We denote by $\mathrm{Ker}(R) \ := \ R \circ R^{-1} \ := \ \{(p,q) \mid \exists r, \ p \ R \ r \ \wedge \ q \ R \ r\}$ the *kernel* of any binary relation R. If R is a bisimulation then its kernel is a bisimulation. Restricting to reductions we obtain exactly the congruences.

Lemma A.2 *The congruences of a graph G are the kernels of the reductions from G.*

Proof.
Let R be a congruence of G. From Lemma A.1, $[\ ]_{_R} : G \longrightarrow G/R$ is a reduction from G satisfying $\mathrm{Ker}([\ ]_{_R}) = \{(p,q) \mid [p]_{_R} = [q]_{_R}\} = R$.
Conversely let h be a reduction from G to some graph H. So h is a bisimulation from G to H. Hence $\mathrm{Ker}(h) = h \circ h^{-1}$ is a bisimulation on G.
As h is a mapping $\mathrm{Ker}(h)$ is an equivalence. Finally $\mathrm{Ker}(h)$ is a congruence.

□

Given a bisimulation R on G and by induction on $n \geq 0$, the composition n times R^n of R is also a bisimulation on G. Hence the reflexive and transitive closure $R^* = \bigcup\{R^n \mid n \geq 0\}$ is a bisimulation on G. In particular the greatest bisimulation $\equiv_G$ of G is also the greatest congruence of G. It follows that bisimilar graphs are graphs reducible to a same graph.

Proposition A.3 *We have* $G \sim H$ *iff* $G \longrightarrow \circ \longleftarrow H$.

Proof.
If $G \longrightarrow \circ \longleftarrow H$ then $G \sim \circ \sim H$ hence $G \sim H$. Let us show the converse.
Let $R : G \sim H$. So $\mathrm{Ker}(R) = R \circ R^{-1}$ is a bisimulation on G. Hence its reflexive and transitive closure $(\mathrm{Ker}(R))^* = (R \circ R^{-1})^*$ denoted by $(RR^{-1})^*$, is also a bisimulation on G, hence a congruence on G. Similarly $(R^{-1}R)^*$ is a congruence on H. From Lemma A.1 the following relation

$$S := ([\ ]_{(\mathrm{RR}^{-1})^*})^{-1} \circ R \circ [\ ]_{(\mathrm{R}^{-1}\mathrm{R})^*}$$

is a bisimulation from $G/(RR^{-1})^*$ to $H/(R^{-1}R)^*$. Note that

$$\begin{array}{rcll} S([p]_{(RR^{-1})^*}) & = & \{[q']_{(R^{-1}R)^*} \mid p\ (RR^{-1})^* R\ q'\} & \text{by definition of } S \\ & = & \{[q']_{(R^{-1}R)^*} \mid p\ R(R^{-1}R)^*\ q'\} & \text{by assciativity of } \circ \\ & = & \{[p']_{(R^{-1}R)^*} \mid p\ R\ p'\} & \\ & = & [p']_{(R^{-1}R)^*} & \text{for any } p' \text{ such that } p\ R\ p', \end{array}$$

because if $p\ R\ p'$ and $p\ R\ p''$ then $p'\ R^{-1}R\ p''$ hence $[p']_{(R^{-1}R)^*} = [p'']_{(R^{-1}R)^*}$.
So S is a functional bisimulation, i.e. a reduction.
Hence $[\ ]_{(\mathrm{RR}^{-1})^*} \circ S : G \longrightarrow H/(R^{-1}R)^*$ and $[\ ]_{(\mathrm{R}^{-1}\mathrm{R})^*} : H \longrightarrow H/(R^{-1}R)^*$ suit.
Remark 1: By symmetry of G and H, S^{-1} is also functional, hence S is an isomorphism.
Remark 2: If $p\ R\ p'$ then $S([p]_{(RR^{-1})^*}) = [p']_{(R^{-1}R)^*}$, i.e. p and p' reduce to a same vertex along these reductions.
□

Let us give some basic properties on tree reductions.

Lemma A.4 *We have the following properties:*
a) $h : \mathrm{Tree}(G,p) \longrightarrow G/p$ *where* $h(u) = q$ *for every path* u *from* p *to* q *;*
b) *If* $h : G \longrightarrow H$ *and* r *is a root of* G *then* $h(r)$ *is a root of* H *;*
c) *If* $h : G \longrightarrow H$ *then* $\mathrm{Tree}(G,p) \longrightarrow \mathrm{Tree}(H,h(p))$ *for every vertex* p *of* G.

Proof.
i) Consider a graph G, a vertex p, and a mapping h associating the goal $h(u)$ to every path u from p.
Let us show that h is a reduction from $\mathrm{Tree}(G,p)$ to G/p .
By definition h is a surjective mapping. It remains to prove that h is a bisimulation.
Let $u \xrightarrow{a} v$ be an arc of $\mathrm{Tree}(G,p)$. So u is a path from p to $h(u)$.
Furthermore there is a vertex q such that $h(u) \xrightarrow{a} q$ and $u(h(u),a,q) = v$.
So $h(v) = q$ hence h is a simulation.
Conversely let $h(u) \xrightarrow{a} q$ be an arc of G/p .

So u is a path from p to $h(u)$ hence $u \xrightarrow{a} u(h(u),a,q)$ is an arc of $\mathrm{Tree}(G,p)$.
As $h(u(h(u),a,q)) = q$, h^{-1} is also a similation. Finally h is a reduction.
ii) We denote by $\mathrm{Path}(G,p)$ the set of paths of G from p.
Let h be a reduction from a graph G to a graph H, and let p be a vertex of G.
By induction on the length of paths in G of source p, we define a mapping g as follows:

$$\begin{array}{rcl} g(\epsilon) & = & \epsilon \\ g(u(q,a,r)) & = & g(u)(h(q),a,h(r)) \quad \text{for every } u(q,a,r) \in \mathrm{Path}(G,p). \end{array}$$

As h is a bisimulation and by induction on the path lengths, we have

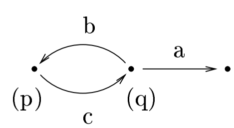
$$\begin{array}{rcl} u \in \mathrm{Path}(G,p) & \Longrightarrow & g(u) \in \mathrm{Path}(H,h(p)), \\ v \in \mathrm{Path}(H,h(p)) & \Longrightarrow & \exists\, u\ ,\ u \in \mathrm{Path}(G,p) \ \wedge\ g(u) = v\ . \end{array}$$

So $p \underset{G}{\longrightarrow}^{*} \circ\, h\ q$ if and only if $h(p) \underset{H}{\longrightarrow}^{*} q$.
In particular $h(p)$ is a root of H when p is a root of G.
iii) Continuing (ii), let us show that g is a reduction from $\mathrm{Tree}(G,p)$ to $\mathrm{Tree}(H,h(p))$.
From (ii) g is a surjective mapping. It remains to prove that g is a bisimulation.
Let $u \xrightarrow{a} v$ be an arc of $\mathrm{Tree}(G,p)$. There is an arc $q \xrightarrow{a} r$ of G such that $u(q,a,r) = v$.
In particular u is a path ending by q. From (ii), $g(u) \in \mathrm{Path}(H,h(p))$.
So $g(u)$ ends by $h(q)$. As $h(q) \xrightarrow{a} h(r)$ is an arc of H, $g(u) \xrightarrow{a} g(u)(h(q),a,h(r)) = g(v)$ is an arc of $\mathrm{Tree}(H,h(p))$. Finally g is a simulation.
Conversely let $g(u) \xrightarrow{a} s$ an arc of $\mathrm{Tree}(H,h(p))$. Let q be the goal of u.
So there is t such that $g(u)(h(q),a,t) = s$. In particular $h(q) \xrightarrow{a} t$ is an arc of H.
Hence there is r such that $q \xrightarrow{a} r$ is an arc of G and $h(r) = t$.
So $u \xrightarrow{a} u(q,a,r)$ is an arc of $\mathrm{Tree}(G,p)$ with

$$g(u(q,a,r)) = g(u)(h(q),a,h(r)) = g(u)(h(q),a,t) = s.$$

□

Let us note that the vertices p and q of the following graph:



are not bisimilar but their unfolded trees $\mathrm{Tree}(G,p)$ and $\mathrm{Tree}(G,q)$ are bisimilar. Nevertheless bisimilar vertices means that their unfolded trees are reducible to a common tree.

Proposition A.5 $p \equiv_G q$ *iff* $\exists\, T$ *tree,* $\mathrm{Tree}(G,p) \longrightarrow T \longleftarrow \mathrm{Tree}(G,q)$.

Proof.
i) Let $p \equiv_G q$: There is a bisimulation R on G such that $p\ R\ q$.
Note that $S : G/p \sim G/q$ for $S := \{(s,t) \in R \mid p \underset{G}{\longrightarrow}^{*} s \ \wedge\ q \underset{G}{\longrightarrow}^{*} t\}$.
From Proposition A.3 there is a graph H and reductions $f : G/p \longrightarrow H$ and $g : G/q \longrightarrow H$.
From Remark 2 in proof of Proposition A.3 we may assume that $f(p) = g(q)$. From Lemma A.4 (c), we have
$\mathrm{Tree}(G,p) = \mathrm{Tree}(G/p,p) \longrightarrow \mathrm{Tree}(H,f(p)) = \mathrm{Tree}(H,g(q)) \longleftarrow \mathrm{Tree}(G/q,q) = \mathrm{Tree}(G,q)$
ii) Suppose there is a tree T such that $\mathrm{Tree}(G,p) \longrightarrow T \longleftarrow \mathrm{Tree}(G,q)$.
From Lemma A.4 (b), there is $R : \mathrm{Tree}(G,p) \sim \mathrm{Tree}(G,q)$ with $\epsilon\ R\ \epsilon$.
From Lemma A.4 (a), there is $S : G/p \sim G/q$ with $p\ S\ q$.

Let Id be the identity relation on G. So $T := S \cup Id$ is a bisimulation on G with $p\, T\, q$.
□

The set of paths of G from p is denoted by $\text{Path}(G,p)$. The label $\text{Label}(p,a,q) = a$ of an arc is extended by morphism to paths, and by union to a set of paths. Restricting Proposition A.5 to deterministic graphs, let us prove an extension of Corollary 2.2 .

Lemma A.6 *Given a deterministic graph G, we have the following properties:*
a) $p \equiv_G q$
b) $\text{Tree}(G,p)$ *is isomorphic to* $\text{Tree}(G,q)$
c) $\text{Label}(\text{Path}(G,p)) = \text{Label}(\text{Path}(G,q))$.

Proof.
a) $\Rightarrow$ c) : Let $p \equiv_G q$: there is a simulation R such that pRq.
For every $u \in \text{Label}(\text{Path}(G,p))$ and by induction on $|u|$, we have $u \in \text{Label}(\text{Path}(G,q))$.
So $\text{Label}(\text{Path}(G,p)) \subseteq \text{Label}(\text{Path}(G,q))$ hence the equality by symmetry of $\equiv_G$.
c) $\Rightarrow$ b) : Suppose that $\text{Label}(\text{Path}(G,p)) = \text{Label}(\text{Path}(G,q))$.
As G is deterministic, to every path x of G from p there is a unique path $h(x)$ of G from q with the same label : $\text{Label}(h(x)) = \text{Label}(x)$.
So h is a total function from $\text{Tree}(G,p)$ to $\text{Tree}(G,q)$.
Let us show that h is a simulation: let $x \xrightarrow{a} y$ be an arc of $\text{Tree}(G,p)$.
In particular y is a path of G from p of label $\text{Label}(y) = \text{Label}(x).a$.
Hence $h(y)$ is a path of G from q of label $\text{Label}(h(y)) = \text{Label}(x).a$.
By unicity of $h(x)$, we have $h(y) = h(x).(s,a,t)$ where s,t are respectively the end of paths $h(x), h(y)$.
By definition of $\text{Tree}(G,q)$ it follows that $h(x) \xrightarrow{a} h(y) \in \text{Tree}(G,q)$.
Similarly to every path y of G from q there is a unique path x of G from p with $\text{Label}(x) = \text{Label}(y)$, hence $y = h(x)$.
By symmetry h^{-1} is also a total function and a simulation.
Finally h is a bijective bisimulation, i.e. an isomorphism.
b) $\Rightarrow$ a) : Suppose that $\text{Tree}(G,p)$ and $\text{Tree}(G,q)$ are isomorphic.
Any isomorphism is a reduction hence $\text{Tree}(G,p) \longrightarrow \text{Tree}(G,q) \longleftarrow \text{Tree}(G,q)$.
From Proposition A.5, $p \equiv_G q$.
□

We say that a vertex p is *terminal* if it is source of no arc (its out-degree is 0). We denote by $\text{End\_Path}(G,p)$ the set of paths of G from p and ending by a terminal vertex. Recall that a graph G is *locally finite* if $\text{End\_Path}(G,p) \neq \emptyset$ for every p.

Lemma A.7 *Given a locally finite graph G,* $\text{Label}(\text{Path}(G,p)) = \text{Label}(\text{Path}(G,q))$ *if and only if* $\text{Label}(\text{End\_Path}(G,p)) = \text{Label}(\text{End\_Path}(G,q))$.

Proof.
The sufficient condition is immediate. Let us prove the necessary condition.
Suppose that G is a locally finite graph.
Consider vertices p and q such that $\text{Label}(\text{End\_Path}(G,p)) = \text{Label}(\text{End\_Path}(G,q))$.

Let x be a path from p to a vertex r.
As G is locally finite, there is a path y from r to a terminal vertex.
So $\text{Label}(x).\text{Label}(y) \in \text{Label}(\text{End\_Path}(G,p)) = \text{Label}(\text{End\_Path}(G,q))$.
In particular there is a path from q of label $\text{Label}(x)$.
So $\text{Label}(\text{Path}(G,p)) \subseteq \text{Label}(\text{Path}(G,q))$ hence the equality by symmetry of p, q.
□

The pruning operation is a particular reduction.

Lemma A.8 *If* $S \geq T$ *then* $S \longrightarrow T$.

Proof.
Let $S \geq T$: there is a set P of nodes of S such that
for every $p \in P$ there are $q \notin P$ and an isomorphism $h(p) : S_p \sim S_q$,
and T is isomorphic to $S - \bigcup\{S_p \mid p \in P\}$. So
$\bigcup\{h(p) \mid p \in P \wedge \neg(\exists q \in P, q \longrightarrow^+ p)\} \cup \{(q,q) \mid \neg(\exists p \in P, p \longrightarrow^* q)\}$
is a reduction from S to $S - \bigcup\{S_p \mid p \in P\}$, hence from S to T.
□

We are ready to characterize bisimilar vertices by taking their unfolded trees with the pruning operation.

Theorem 2.4 *We have the following equivalences:*
$p \equiv_G q$ *iff* $\text{Tree}(G,p)\ (\geq \cup \leq)^*\ \text{Tree}(G,q)$
iff $\text{Tree}(G,p) \leq T \geq \text{Tree}(G,q)$ *for some tree* T.

Proof.
i) If $\text{Tree}(G,p) \leq \circ \geq \text{Tree}(G,q)$ then $\text{Tree}(G,p)\ (\geq \cup \leq)^*\ \text{Tree}(G,q)$.
ii) If $\text{Tree}(G,p)\ (\geq \cup \leq)^*\ \text{Tree}(G,q)$ then by Lemma A.8 and Lemma A.4 (b),
there is $R : \text{Tree}(G,p) \sim \text{Tree}(G,q)$ with $\epsilon\ R\ \epsilon$.
From Lemma A.4 (a), there is $S : G/p \sim G/q$ with $p\ S\ q$.
So $T := S \cup Id$ is a bisimulation on G with $p\ T\ q$, i.e. $p \equiv_G q$.
iii) Let $p \equiv_G q$. It remains to prove that $\text{Tree}(G,p) \leq \circ \geq \text{Tree}(G,q)$.
From Proposition A.5 and Lemma A.4 (b), there is $R : \text{Tree}(G,p) \sim \text{Tree}(G,q)$ with $\epsilon\ R\ \epsilon$.
So the following disjoint union H of $\text{Tree}(G,p)$ and $\text{Tree}(G,q)$
$H := \{(p,u) \xrightarrow{a} (p,v) \mid u \xrightarrow{a} v \in Tree(G,p)\} \cup \{(q,u) \xrightarrow{a} (q,v) \mid u \xrightarrow{a} v \in Tree(G,q)\}$
satisfies $(p,\epsilon) \equiv_H (q,\epsilon)$ and has *no cycle*: there is no nonempty path from any vertex to itself, i.e. $p \neq q$ for $p \longrightarrow^+ q$.
More generally consider a graph G with no cycle, and a set P of bisimilar vertices of G, i.e. $p \equiv_G q$ for every $p, q \in P$, and let us show that there is a tree T such that $T \geq \text{Tree}(G,p)$ for every $p \in P$.
We denote by $V := \{q \mid \exists p \in P,\ p \longrightarrow^* q\}$ the set of vertices accessible from P.
Let $d(P,q) := min\{n \mid \exists p \in P, p \longrightarrow^n q\}$ be the distance from P to q in V, i.e. the minimum length of paths from a vertex of P to q.
We restrict $\equiv_G$ to vertices in V having the same distance from P:

$p \sim q \quad \text{if} \quad p \equiv_G q \ \wedge \ d(P,p) = d(P,q).$

As G has no cycle, $\sim$ is a bisimulation. Let $[p] := \{q \mid q \sim p\}$ be the equivalence class of $p \in V$ according to $\sim$. In particular $P = [p]$ for every p in P. Let us define

$$H := \{ ([p],u) \xrightarrow{a} ([q],u(p,a,q)) \mid p \in V \ \wedge \ (p,a,q) \in G \ \wedge \ u \in G^* \}$$
$$\text{and} \quad T := H/(P,\epsilon).$$

Then T is a tree: (P,ϵ) is a root and is goal of no arc; furthermore $([q],u(p,a,q))$ is goal of the unique arc labelled by a with source $([p],u)$.

Let $r \in P$ and let us show that $T \geq \mathrm{Tree}(G,r)$.

Let us take the most outside nodes of T such that their second component is not a path from r, i.e.

$Q := \{ ([q],v) \mid \exists\, p,u,a,\ ([p],u) \xrightarrow{a} ([q],v) \ \in T \ \wedge \ u \in Path(G,r) \ \wedge \ v \notin Path(G,r) \}$.

Then $T - \bigcup\{ T_u \mid u \in Q \} = \{ ([p],u) \xrightarrow{a} ([q],u(p,a,q)) \ \in T \mid u(p,a,q) \in Path(G,r) \}$, hence is isomorphic to $\mathrm{Tree}(G,r)$ according to the second projection.

Let $([q],v) \in Q$. So there is an arc $([p],u] \xrightarrow{a} ([q],v)$ in T such that u is a path from r to a vertex s, and v is not a path from r. By definition of T and H, $v = u(p,a,q)$, $p \xrightarrow{a} q$ is an arc of G, $s \neq p$ but $s \sim p$.

As $\sim$ is a bisimulation, there is t such that $s \xrightarrow{a} t$ is an arc of G with $t \sim q$.

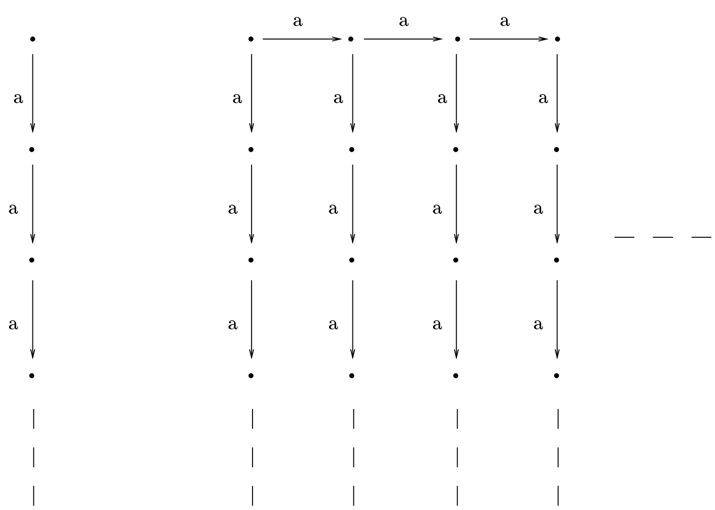
So $([s],u) \xrightarrow{a} ([t],u(s,a,t))$ is a transition of T and $([t],u(s,a,t)) \notin Q$.

As $T/([p],u) = H/([p],u)$ is a tree independant of u for every p and u,

the tree $T/([t],u(s,a,t)) = T/([q],u(s,a,t))$ is isomorphic to $T/([q],v)$. Finally $T_{([q],v)}$ is isomorphic to $T_{([t],u(s,a,t))}$ with the same root $([p],u)$.

□

Let us notice that these two non isomorphic trees



have bisimilar roots but are irreducible along the pruning reduction. Proposition A.3 and Proposition A.5 remain true if we inverse the reductions.

Proposition A.9 *We have the following properties:*

a) $p \equiv_G q$ *iff* $\exists\, T$ *tree,* $\mathrm{Tree}(G,p) \longleftarrow T \longrightarrow \mathrm{Tree}(G,q)$.

b) $G \sim H$ *iff* $G \longleftarrow \circ \longrightarrow H$.

Proof.
The necessary condition of (a) follows from Lemma A.8 and Theorem 2.4 .
The sufficient condition of (a) is similar to the sufficient condition of Proposition A.5 .
If $G \longleftarrow \circ \longrightarrow H$ then $G \sim \circ \sim H$ hence $G \sim H$. Let us show the converse. Before let us prove an intermediate result.
i) Let $(h_i \, : \, G_i \longrightarrow H_i)_{i \in I}$ be a set of reductions such that for every vertex p common to H_i and H_j , we have $H_i/p \; = \; H_j/p$.
Let G be the disjoint union of the $(G_i)_{i \in I}$, i.e.

$$G \;:=\; \{ \; (i,u) \xrightarrow{a} (i,v) \mid i \in I \; \wedge \; u \xrightarrow{a} v \; \in G_i \; \} \; .$$

Let $h(i,u) = h_i(u)$ for every $i \in I$ and every vertex u of G_i .
Let us show that h is a reduction from G to $H \; := \; \bigcup \{ H_i \mid i \in I \}$. Note that h is total and onto.
Consider an arc $(i,u) \xrightarrow{a} (i,v)$ of G. By definition of G, $u \xrightarrow{a} v$ is an arc of G_i. As h_i is a reduction $h(i,u) \; = \; h_i(u) \xrightarrow{a} h_i(v) \; = \; h(i,v)$ is an arc of H_i hence of H. Thus h is a simulation.
Conversely consider an arc $h(i,u) \xrightarrow{a} w$ of H. There is $j \in I$ such that $h(i,u) \xrightarrow{a} w$ is an arc of H_j. So $h(i,u)$ is a common vertex of H_i and H_j.
By hypothesis $H_i/h(i,u) \; = \; H_j/h(i,u)$ hence $h(i,u) \xrightarrow{a} w$ is an arc of H_i.
As $h_i(u) = h(i,u)$ and h_i is a reduction, there is an arc $u \xrightarrow{a} v$ of G_i such that $h_i(v) = w$.
By definition of G, $(i,u) \xrightarrow{a} (i,v)$ is an arc of G with $h(i,v) \; = \; h_i(v) \; = \; w$.
ii) Given a graph G, we consider the disjoint union Unfold(G) of its unfolded trees:

$$\text{Unfold}(G) \;:=\; \{ \; (p,u) \xrightarrow{a} (p,v) \mid u \xrightarrow{a} v \; \in \text{Tree}(G,p) \; \} \; .$$

From (i) and Lemma A.4 (a), we have Unfold$(G) \longrightarrow G$.
Let $R : G \sim H$. By Theorem 2.4 and Lemma A.8, for each pRq there is a tree $T_{(p,q)}$ such that

$$\text{Tree}(G,p) \longleftarrow T_{(p,q)} \longrightarrow \text{Tree}(G,q).$$

Let K be the disjoint union of the trees $T_{(p,q)}$ i.e.

$$K \;:=\; \{ \; (p,q,u) \xrightarrow{a} (p,q,v) \mid pRq \; \wedge \; u \xrightarrow{a} v \; \in \text{Tree}_{(p,q)} \; \} \; .$$

Using (i) we obtain

$$G \longleftarrow \text{Unfold}(G) \longleftarrow K \longrightarrow \text{Unfold}(H) \longrightarrow H \; .$$

□

A.b Pushdown automata and regular graphs

Let G be a graph with a vertex r and let f be a function from the vertices of G to the vertices of another graph. We denote by

G_r the connected component of G containing r ,
$G^{-1} \; := \; \{ \; v \xrightarrow{a} u \mid (u \xrightarrow{a} v) \; \in G \; \}$ the *inverse* of G ,
$f(G) \; := \; \{ \; f(u) \xrightarrow{a} f(v) \mid (u \xrightarrow{a} v) \; \in G \; \}$ the *image* of G by f .

We give conditions on f in such a way that f preserves the accessible subgraphs and the connected components of G.

Lemma A.10 *Let f be a function from the vertices of a graph G into the vertices of a graph H. If f and f^{-1} are partial simulations then*

$$f(G/r) \;=\; f(G)/f(r) \;=\; H/f(r) \quad \text{for any } r \text{ in the domain of } f.$$

Furthermore if f^{-1} is also a partial simulation from H^{-1} into G^{-1} then

$$f(G_r) \;=\; f(G)_{f(r)} \;=\; H_{f(r)} \quad \text{for any } r \text{ in the domain of } f.$$

Proof.
Let r be in the domain of f.
If f is a partial simulation then by induction on $n \geq 0$, we have for any vertex u of G:

$$r \underset{G}{\longrightarrow}^n u \implies f(r) \underset{H}{\longrightarrow}^n f(u) \quad \text{and} \quad r \underset{G}{\longleftrightarrow}^n u \implies f(r) \underset{H}{\longleftrightarrow}^n f(u) \ .$$

Hence $f(G/r) \subseteq f(G)/f(r)$ and $f(G_r) \subseteq f(G)_{f(r)}$.
If f^{-1} is a partial simulation then by induction on $n \geq 0$, we have for any vertex w of H:

$$f(r) \underset{H}{\longrightarrow}^n w \implies \exists\, v\ ,\ r \underset{G}{\longrightarrow}^n v \ \wedge\ f(v) = w \ .$$

Hence $H/f(r) \subseteq f(G/r)$.
If f^{-1} is a partial simulation from H into G, and from H^{-1} into G^{-1}, then by induction on $n \geq 0$, we have for any vertex w of H:

$$f(r) \underset{H}{\longleftrightarrow}^n w \implies \exists\, v\ ,\ r \underset{G}{\longleftrightarrow}^n v \ \wedge\ f(v) = w \ .$$

Hence $H_{f(r)} \subseteq f(G_r)$.
As $f(G) \subseteq H$, it follows the equalities.
□

Let us show that the pushdown automata and the word rewriting systems have the same connected structures by prefix rewriting.

Proposition 4.1 *The prefix transition graphs of labelled word rewriting systems and the pushdown transition graphs have effectively the same connected components and the same accessible subgraphs.*

Proof.
Let T be the set of terminals. Let R be a rewriting system on a set M of non-terminals, and let $r \in M^*$.
We recall the construction in [Ca 90 b] of a pushdown automaton P on a set $\overline{N}$ of non-terminals and on a set Q of states, and a configuration $c \in Q.\overline{N}$ such that the accessible subgraph $(P.\overline{N}^*)/c$ from c of the transition graph of P is isomorphic to the accessible subgraph $(R.M^*)/r$ from r of the prefix transition graph of R. Then we prove also that the connected component $(P.\overline{N}^*)_c$ containing c of $P.\overline{N}^*$ is isomorphic to $(R.M^*)_r$.
We may suppose that $r \neq \epsilon$ and that R is *ϵ-free*: for every rule $u \overset{a}{\longrightarrow} v$ of R, both u and v are nonempty. Otherwise we could add a new symbol \$ in M and replace (R, r) by $(\$R, \$r)$ where $\$R := \{ \$u \overset{a}{\longrightarrow} \$v \mid (u \overset{a}{\longrightarrow} v) \in R \}$. It suffices to note that

$$(\$R.(M{\cup}\{\$\})^*)/\$r \;=\; (\$R.M^*)/\$r \;=\; \$(R.M^*)/\$r \;=\; \$((R.M^*)/r)$$

$$\text{and} \quad (\$R.(M{\cup}\{\$\})^*)_{\$r} \;=\; (\$R.M^*)_{\$r} \;=\; (\$(R.M^*))_{\$r} \;=\; \$((R.M^*)_r) \ .$$

We say that R is *normal* if for every rule $u \overset{a}{\longrightarrow} v$ of R, both u and v have length (strictly) smaller than 3.

i) First we transform (M, R, r) into another 3-uple (N, S, s) preserving the accessible

subgraph and the connected component of $R.M^*$ from r, and such that S is normal and ϵ-free, and s is a letter.
Let m be the greatest length of r and the words in R, i.e.

$$m \ := \ max\{ \ |u| \ | \ (u = r) \ \vee \ \exists a \ \exists v \ ((u \xrightarrow{a} v) \in R \ \vee \ (v \xrightarrow{a} u) \in R) \ \} \ .$$

We take an injection i from $\{u \in M^+ \mid 1 \leq |u| \leq m\}$ to some given alphabet N. Then we extend i to an injection j from M^* into N^* defined by induction as follows:

$$j(\epsilon) \ := \ \epsilon \ ,$$
$$j(u) \ := \ j(v)i(w) \qquad \text{where} \ \ u = vw \neq \epsilon \ \wedge \ |w| = min(m, |u|).$$

The rewriting system S on N is defined by

$$S \ := \ \{ \ j(uw) \xrightarrow{a} j(vw) \mid (u \xrightarrow{a} v) \in R \ \wedge \ w \in M^* \ \wedge \ |w| < m \ \} \ .$$

Note that S is normal and ϵ-free.
Furthermore if R is a context-free grammar then S is also a context-free grammar.
Let us prove that S and the letter $s = j(r)$ suit.

i1) Let us show that j is a simulation from $R.M^*$ into $S.N^*$.
Let $u \xrightarrow{a} v$ be in $R.M^*$. There exist a rule $u_0 \xrightarrow{a} v_0$ of R and a word $w \in M^*$ such that $u = u_0w$ and $v = v_0w$. Consider the decomposition $w = xy$ where y is the greatest suffix of w such that its length $|y|$ is a multiple of m. So

$$j(u) \ = \ j(u_0w) \ = \ j(u_0xy) \ = \ j(u_0x)j(y)$$
$$\text{and} \quad j(v) \ = \ j(v_0w) \ = \ j(v_0xy) \ = \ j(v_0x)j(y) \ .$$

By definition of y, we have $|x| < m$. Hence $j(u_0x) \xrightarrow{a} j(v_0x)$ belongs to S.
Consequently $j(u) \xrightarrow{a} j(v)$ is a transition of $S.N^*$.

i2) Let us show that j^{-1} is a partial simulation from $S.N^*$ into $R.M^*$.
Let $j(u) \xrightarrow{a} w$ be in $S.N^*$. There exist a rule $p \xrightarrow{a} q$ of S and a word $t \in N^*$ such that $j(u) = pt$ and $w = qt$. By definition of S, there is a rule $u_0 \xrightarrow{a} v_0$ of R and a word w_0 of M^* such that $|w_0| < m$, $p = j(u_0w_0)$ and $q = j(v_0w_0)$.
As R is ϵ-free, $p \neq \epsilon$. Furthermore $pt \in Im(j)$. So there is $x \in M^*$ such that $j(x) = t$ and $|x| = m|t|$.

$$\text{Thus} \quad j(u) \ = \ pt \ = \ j(u_0w_0)j(x) \ = \ j(u_0w_0x)$$
$$\text{and} \quad w \ = \ qt \ = \ j(v_0w_0)j(x) \ = \ j(v_0w_0x).$$

As j is injective, we have $u = u_0w_0x$. So $v = v_0w_0x$ suits: $u \xrightarrow{a} v$ is a transition of $R.N^*$ and $j(v) = w$.

i3) Similarly to (i2) and given a transition $w \xrightarrow{a} j(v)$ of $S.N^*$, there is a transition $u \xrightarrow{a} v$ of $R.M^*$ such that $j(u) = w$. This means that j^{-1} is a partial simulation from $(S.N^*)^{-1}$ into $(R.M^*)^{-1}$.
As j is an injective application and by (i1), (i2) and Lemma A.10, $(R.M^*)/r$ is isomorphic to $(S.N^*)/s$ and $(R.M^*)_r$ is isomorphic to $(S.N^*)_s$.

ii) It remains to transform (S, s) into a couple (P, c) of a pushdown automaton P on a non-terminal set $\overline{N}$, and a configuration c such that $(P.\overline{N}^*)/c$ is isomorphic to $(S.N^*)/s$ and $(P.\overline{N}^*)_c$ is isomorphic to $(S.N^*)_s$.
We define the following system:

$$\overline{S} \ := \ \{ \ ux \xrightarrow{a} vx \mid (u \xrightarrow{a} v) \in S \ \wedge \ x \in \overline{N}^* \ \wedge \ |ux| = 2 \ \}$$

where $\overline{N} \ := \ N \ \cup \ \{\&\}$ with & is a new symbol.
Note that the vertices of $(S.\overline{N}^*)_{s\&}$ belong to $N^+.\&$, hence of length ≥ 2. It follows the above equalities:

$$(\overline{S}.\overline{N}^*)/s\& \ = \ (S.\overline{N}^*)/s\& \ = \ (S.N^*.\&)/s\& \ = \ ((S.N^*)/s).\&$$

and $$(\overline{S}.\overline{N}^*)_{s\&} \ = \ (S.\overline{N}^*)_{s\&} \ = \ (S.N^*.\&)_{s\&} \ = \ (S.N^*)_s.\& \ .$$

Let us denote by

$$N' \ := \ \{ \ u(1) \mid u = s \ \vee \ \exists \, a \, \exists \, v \ ((u \xrightarrow{a} v) \in S \ \vee \ (v \xrightarrow{a} u) \in S) \ \}$$

the set of the first letters of s and S (or $\overline{S}$).
We take an injection f from N' to an alphabet Q disjoint of $\overline{N}$. We extend f to a total injection from $N'.\overline{N}^*$ to $Q.\overline{N}^*$ as follows:

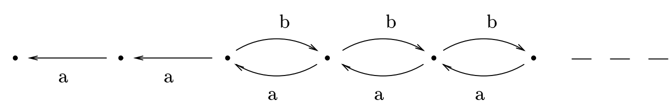
$$f(au) \ := \ f(a).u \quad \text{for every } a \in N' \text{ and } u \in \overline{N}^*.$$

Let us show that the pushdown automaton $P := f(\overline{S})$ and the configuration $c \ := \ f(s).\&$ suit. We have

$$(P.\overline{N}^*)/c \ = \ (f(\overline{S}).\overline{N}^*)/f(s)\& \ = \ f(\overline{S}.\overline{N}^*)/f(s\&) \ = \ f((\overline{S}.\overline{N}^*)/s\&) \ = \ f(((S.N^*)/s).\&)$$

and similarly $(P.\overline{N}^*)_c \ = \ f((S.N^*)_s.\&)$.
Finally $(P.\overline{N}^*)/c$ is isomorphic to $(S.N^*)/s$ hence to $(R.M^*)/r$ and $(P.\overline{N}^*)_c$ is isomorphic to $(S.N^*)_s$ hence to $(R.M^*)_r$.
□

For instance the system $R \ = \ \{x \xrightarrow{a} \epsilon \ , \ x^2 \xrightarrow{b} x^3\}$ on the non-terminal set $\{x\}$ has the following prefix transition graph $R.\{x\}^*$:



and is accessible from $r = x^2$ (or any word in x^2x^+). Following the proof of Proposition 4.1, the system R is transformed into $\$R = \{\$x \xrightarrow{a} \$ \ , \ \$x^2 \xrightarrow{b} \$x^3\}$ and the axiom r is replaced by $\$r = \$x^2$.
Associating $\$x$ to x, $\$x^2$ to y, $\$x^3$ to z, x^4 to s and $ to $, $R.\{x\}^*$ is the accessible subgraph from y and is the connected component containing y of the following system:

$$S \ = \ \{x \xrightarrow{a} \$ \ , \ y \xrightarrow{a} x \ , \ z \xrightarrow{a} y \ , \ \$s \xrightarrow{a} z \ ,$$
$$y \xrightarrow{b} z \ , \ z \xrightarrow{b} \$s \ , \ \$s \xrightarrow{b} xs \ , \ xs \xrightarrow{b} ys \ \}.$$

The set $\text{Path}(G,p,q)$ of pathes of a graph G from a vertex p to a vertex q is extended by union to the set $\text{Path}(G,p,V)$ of pathes from p to any vertex in V, i.e.

$$\text{Path}(G,p,V) \ := \ \bigcup \{ \ \text{Path}(G,p,q) \mid q \in V \ \} \ .$$

Let P be a finite subset of $Q.N \times T \times \ Q.N^*$, i.e. P is a real-time pushdown automaton on a set T of terminals, a set Q of states and a set N of stack letters (or non-terminals). Recall that a *configuration* is an element of $Q.N^*$. Given a configuration r and a set V of configurations, we have

$$L(P.N^*,r,V) \ = \text{Label}(\text{Path}(P.N^*,r,V)) \ .$$

If P is deterministic and reduced, then $P.N^*$ is deterministic and is locally finite: every terminal vertex is a state, so we have

$$
\begin{aligned}
u \equiv_{P.N^*} v \quad & \text{iff} \quad \text{Label}(\text{Path}(P.N^*,u)) = \text{Label}(\text{Path}(P.N^*,v)) \quad \text{by Lemma A.6} \\
& \text{iff} \quad \text{Label}(\text{End\_Path}(P.N^*,u)) = \text{Label}(\text{End\_Path}(P.N^*,v)) \quad \text{by A.7} \\
& \text{iff} \quad \text{Label}(\text{Path}(P.N^*,u,Q)) = \text{Label}(\text{Path}(P.N^*,v,Q)) \\
& \text{iff} \quad L(P.N^*,u,Q) = L(P.N^*,v,Q)\ .
\end{aligned}
$$

Thus the bisimulation of any reduced real-time dpda P is inter-reducible to the equivalence problem of P with acceptance on empty stack, and this problem is decidable [OIH 80].
If P is deterministic then $P.N^*$ is deterministic and we have

$$
\begin{aligned}
u \equiv_{P.N^*} v \quad & \text{iff} \quad \text{Label}(\text{Path}(P.N^*,u)) = \text{Label}(\text{Path}(P.N^*,v)) \quad \text{by Lemma A.6} \\
& \text{iff} \quad \text{Label}(\text{Path}(P.N^*,u,Q.N^*)) = \text{Label}(\text{Path}(P.N^*,v,Q.N^*)) \\
& \text{iff} \quad L(P.N^*,u,Q.N^*) = L(P.N^*,v,Q.N^*)\ .
\end{aligned}
$$

Thus the bisimulation of any real-time dpda P is inter-reducible to the equivalence problem of P with acceptance on any state (any configuration), i.e. is reducible to the decidable equivalence problem of P with acceptance on any final state [Ro 86].

Let us give some notations. Given a binary relation R, we denote by $Dom(R) := \{x \mid \exists y,\ xRy\}$ the *domain* of R, and by $Im(R) := \{y \mid \exists x,\ xRy\}$ the *image* (or range) of R. Recall that $u(i)$ is the letter at the i-th occurrence of a word u, and $|u|$ is the length of u. We denote by $V_{as_1 \ldots s_p} := \{s_1, \ldots, s_p\}$ the set of vertices of any hyperarc $as_1 \ldots s_p$, and by $V_H := \bigcup\{V_X \mid X \in H\}$ the set of vertices of any hypergraph H. We consider the set

$$R^\omega(H) \ := \ \{\ \textstyle\bigcup_{n \geq 0}[H_n] \ \mid\ H_0 = H \ \wedge\ \forall\, n \geq 0,\ H_n \underset{R}{\Longrightarrow} H_{n+1}\ \}$$

of (isomorphic) graphs generated by a deterministic graph grammar R from a hypergraph H, and preserving H.
Let us give some basic transformations of deterministic graph grammars.
A deterministic graph grammar R is *reduced* from a hypergraph G if every non-terminal of R is accessible from G:

$$\forall X \in Dom(R)\ \ \exists H\ \exists Y\ ,\ G \underset{R}{\Longrightarrow}^* H \ \wedge\ Y \in H \ \wedge\ Y(1) = X(1)\ .$$

We can put every deterministic graph grammar in a reduced form.

Lemma A.11 *Any pair (R,G) of a deterministic graph grammar R and a finite hypergraph G may be effectively transformed into another pair (S,H) such that S is reduced from H and $S^\omega(H) = R^\omega(G)$.*

Proof.
Let R be a deterministic graph grammar and G be a finite hypergraph.
Let $N := \{X(1) \mid X \in Dom(R)\}$ be the set of non-terminals of R.
As usual, the set Acc of accessible non-terminals from G is computable: it is the least subset of N containing the non-terminals of G, and closed by the following accessibility relation:

$$\{\ (X(1), Y(1)) \mid \exists H,\ (X,H) \in R \ \wedge\ Y \in H \ \wedge\ Y(1) \in N\ \}$$

More exactly Acc is the least fixpoint of the following equation:

$$Acc \ = \ \{\ X(1) \in N \mid X \in G\ \} \cup \{\ Y(1) \in N \mid \exists\,(X,H) \in R,\ X(1) \in Acc \wedge Y \in H\ \}.$$

To remove the hyperarcs labelled by a non-terminal not in Acc, we define the following deterministic graph grammar:

$$I := \{(X, \emptyset) \mid X \in Dom(R) \ \wedge\ X(1) \notin Acc\}\ .$$

So the following graph grammar:

$$S := \{(X,K) \mid \exists H,\ (X,H) \in R \ \wedge\ X(1) \in Acc \ \wedge\ H \underset{I}{\Longrightarrow} K\ \}$$

and the unique hypergraph H defined by $G \underset{I}{\Longrightarrow} H$ suit:
S is reduced from H and $S^{\omega}(H) = R^{\omega}(G)$.
□

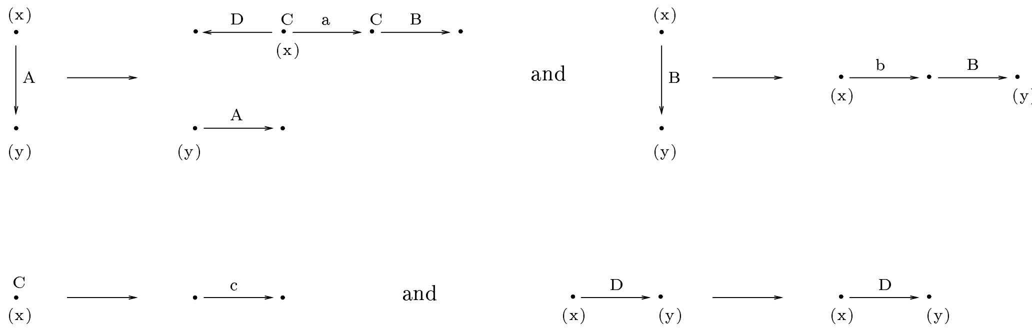
A deterministic graph grammar R is *proper* if for all rule (X,H) of R, every vertex of X is a vertex of a terminal arc of H. We can put every deterministic graph grammar in a reduced proper form.

Lemma A.12 *Any pair (R,G) of a deterministic graph grammar R and a finite hypergraph G may be effectively transformed into another pair (S,H) such that S is proper and $S^{\omega}(H) = R^{\omega}(G)$.*

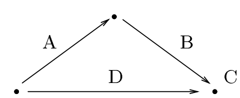
Proof.
Let R be a deterministic graph grammar and G be a finite hypergraph. Let
$N \ := \ \{ X(1) \mid X \in Dom(R) \}$ be the set of non-terminals of R, and
$T \ := \ \{ X(1) \mid \exists H \in Im(R),\ X \in H \ \wedge\ X(1) \notin N \}$ be the set of terminals of R.
i) We remove in R and G the useless non-terminal hyperarcs.
We consider the set $Acc := \{X(1) \mid X \in Dom(R) \ \wedge\ R^{\omega}(X) \neq \{\emptyset\} \}$ of non-terminals generating by R a non empty graph. This set is computable ; it is the least subset of N satisfying the following equality:
$$Acc \ = \ \{ X(1) \mid \exists H, \exists Y,\ (X,H) \in R \ \wedge\ Y \in H \ \wedge\ Y(1) \in T \cup Acc \} \ .$$
To remove the hyperarcs labelled by a non-terminal not in Acc, we define the following deterministic graph grammar:
$$I := \{(X,\emptyset) \mid X \in Dom(R) \ \wedge\ X(1) \notin Acc\} \ .$$
So the grammar $R_1 := \{(X,K) \mid \exists H,\ (X,H) \in R \ \wedge\ X(1) \in Acc \ \wedge\ H \underset{I}{\Longrightarrow} K \}$ and the unique hypergraph G_1 defined by $G \underset{I}{\Longrightarrow} G_1$ satisfy ${R_1}^{\omega}(G_1) = R^{\omega}(G)$ and R_1 is reduced from G_1.
ii) We remove in R_1 and G_1 the useless vertices of non-terminal hyperarcs.
To every $X \in Dom(R_1)$ we associate its set $Acc(X)$ of *useful vertices*, i.e. of vertices of X which are vertices of any graph in ${R_1}^{\omega}(X)$. These sets $Acc(X)$ for $X \in Dom(R_1)$ are computable as being the least fixpoints of the following system:
$$Acc(X) \ = \ \{ s \in V_X \mid \exists Y \in H,\ Y(1) \in T \ \wedge\ s \in V_Y \} \ \cup$$
$$\bigcup \{ \{Y(i) \in V_X \mid Z(i) \in Acc(Z)\} \mid Y \in H \ \wedge\ Z \in Dom(R_1) \ \wedge\ Y(1) = Z(1) \}$$
for every rule $X \longrightarrow H$ of R_1.
For every $X \in Dom(R_1)$ and by definitions of R_1 and $Acc(X)$, we can construct a hypergraph H_X such that
$$X \ R_1 \circ \underset{R_1}{\longrightarrow}^* H_X \ \wedge\ [H_X] \neq \emptyset \ \wedge\ Acc(X) \subseteq V_{[H_X]} \ .$$
To each $X \in Dom(R_1)$ we associate a hyperarc $\overline{X}$ labelled by a new symbol $\overline{X}(1)$ with $\overline{X}(1) \neq \overline{Y}(1)$ for $X \neq Y$, and with distinct vertices such that $V_{\overline{X}} = Acc(X)$ if $Acc(X) \neq \emptyset$, and $V_{\overline{X}}$ is reduced to a vertex of $[H_X]$ if $Acc(X) = \emptyset$. As in (i) to remove the useless vertices of non-terminal hyperarcs, we define the following deterministic graph grammar:
$$J := \{ \ (X,\overline{X}) \mid X \in Dom(R_1) \ \} \ .$$
To each $X \in Dom(R_1)$ we associate a hyperarc $H_{\overline{X}}$ such that $H_X \underset{J}{\Longrightarrow} H_{\overline{X}}$.
So $S \ = \ \{ \ (\overline{X}, H_{\overline{X}}) \mid X \in Dom(R_1) \ \}$ and any hypergraph H defined by $G_1 \underset{J}{\Longrightarrow} H$ suit:

S is proper and $S^{\omega}(H) = {R_1}^{\omega}(G_1) = R^{\omega}(G)$.
□

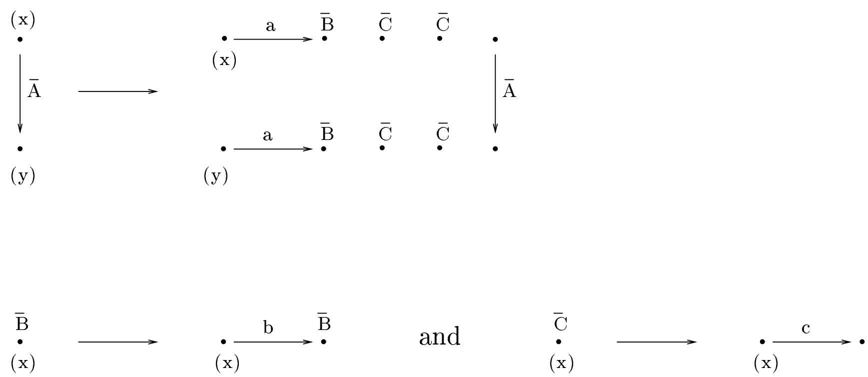
Let us apply Lemma A.12 to the following deterministic graph grammar R:



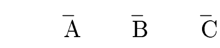
and to the following hypergraph G:



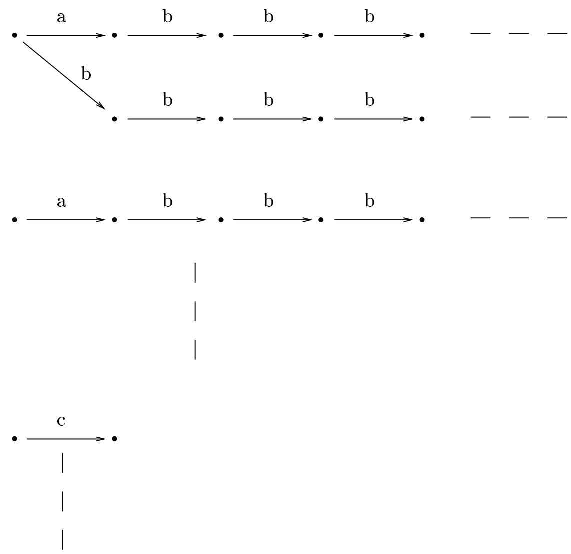
We have $Acc = \{A, B, C\}$, $Acc(A) = \{x, y\}$, $Acc(B) = \{x\}$, $Acc(C) = \emptyset$. By Lemma A.12 we obtain the following proper deterministic graph grammar S:



and the following hypergraph H:



such that $S^{\omega}(H) = R^{\omega}(G)$ and is represented as follows:



Let us give others effective transformations.

Proposition 4.5 *We have the following properties:*

a) *Every accessible subgraph of a regular graph is effectively a regular graph;*

b) *Every connected component of a regular graph is effectively a regular graph;*

c) *Every regular graph has effectively a finite number of nonisomorphic connected components.*

Proof.

i) Consider a deterministic graph grammar R, a finite hypergraph G and a vertex v of G. We will construct a deterministic graph grammar S, a finite hypergraph H such that $S^{\omega}(H) = R^{\omega}(G)/v$.

We denote by N and T the respective sets of non-terminals and terminals of R. We take a new label a, i.e. $a \notin T \cup N$, and we add to R the rule $av \longrightarrow G$.

After a possible renaming (and adding new rules) we may suppose that every hypergraph in $Im(R)$ does not have two non-terminal hyperarcs with the same label. For instance the grammar reduced to the unique rule $Axy \longrightarrow \{axs, ayt, ayz, Ast, Asz\}$ is transformed into the following grammar

$$\{(Axy, \{axs, ayt, ayz, Ast, Bsz\}), (Bxy, \{axs, ayt, ayz, Ast, Bsz\})\} \ .$$

For every $X \in Dom(R)$ and for every $I \subseteq \{2, \ldots, |X|\}$, we associate the set $Acc(X, I)$ of vertices in $E := \{X(i) \mid i \in I\}$ and of vertices in K/E for all K in $R^{\omega}(X)$ restricted to the vertices accessible from E. These sets $Acc(X, I)$ are computable as being the least fixpoints of the following system:

$$\begin{aligned} Acc(X,I) \ = \ & \{X(i) \mid i \in I\} \cup \{\ Y(3) \mid Y \in H \ \wedge \ Y(1) \in T \ \wedge \ Y(2) \in Acc(X,I)\ \} \ \cup \\ & \bigcup \{\ \{Y(i) \mid Z(i) \in Acc(Z, \{j | Y(j) \in Acc(X,I)\})\} \mid Y \in H \ \wedge \ Z \in Dom(R) \\ & \wedge \ Y(1) = Z(1)\ \} \end{aligned}$$

for every rule $X \longrightarrow H$ of R and $I \subseteq \{2, \ldots, |X|\}$.

For every $X \in Dom(R)$, $I \subseteq \{2, \ldots, |X|\}$ and every hyperarc Y satisfying the following condition:

$$C(Y,X,I) \ := \ (V_Y \cap Acc(X,I) \neq \emptyset) \ ,$$

we define the following hyperarc:

$f_{(X,I)}(Y) \ := \ Y(1).g_{(X,I)}(Y(2)).g_{(X,I)}(Y(3)) \quad \text{if } Y(1) \in T \ ,$

$f_{(X,I)}(Y) \ := \ (Y(1), \{i \mid Y(i) \in Acc(X,I)\}).g_{(X,I)}(Y(2))\ldots g_{(X,I)}(Y(|Y|)) \quad \text{if } Y(1) \in N$

where $g_{(X,I)}$ is constant in $Acc(X,I)$, and is the empty word ϵ otherwise.

Let us decompose R according to accessible vertices:

$$S \ := \ \{ \ (f_{(X,I)}(X) \ , \ \{f_{(X,I)}(Y) | Y \in H \wedge C(Y,X,I)\} \mid (X,H) \in R \ \wedge \ C(X,X,I) \ \} \ .$$

Note that S is not necessarily proper. Let $(a,2)v \longrightarrow H$ be the unique rule of S associated to the non-terminal $(a,2)$. Then $S^{\omega}(H) = R^{\omega}(G)/v$.

ii) Consider a deterministic graph grammar R, a finite hypergraph G and a vertex v of G. Consider a bijection from the set T of terminals to a new set $\overline{T}$, associating to each label a a new label $\overline{a}$. We complete R into the following system:

$$\overline{R} \ := \ \{(X \ , \ H \cup \{s \overset{\overline{a}}{\longrightarrow} t \mid s \overset{a}{\longrightarrow} t \ \in H\} \ \mid \ (X,H) \in R \ \} \ .$$

From (i) we construct a system $\overline{S}$ and a finite hypergraph $\overline{H}$ such that $\overline{S}^{\omega}(\overline{H}) = \overline{R}^{\omega}(G)/v$.

By removing the labels of $\overline{T}$, we obtain the following system:

$$S \ := \ \{ \ (X, H - \{s \overset{a}{\longrightarrow} t \mid a \in \overline{T}\}) \ \mid \ (X,H) \in \overline{S} \ \}$$

and the following hypergraph:

$$H \ := \ \overline{H} - \{s \overset{a}{\longrightarrow} t \mid a \in \overline{T}\}$$

such that $S^{\omega}(H)$ is the connected component of $R^{\omega}(G)$ containing v.

iii) Consider a deterministic graph grammar R and a finite hypergraph G. We will generalize (b) to obtain effectively the connected components of $R^{\omega}(G)$.

By Lemma A.12 we may assume that R is proper. So for every hyperarc X and every vertex v of X, $R^{\omega}(G)$ has a connected component containing v.

As in (i), we suppose that every hyperarc in $Im(R)$ does not have two non-terminal hyperarcs with the same label.

From Lemma A.11 we may assume that R is also reduced.

Similarly to (i), for every rule $X \longrightarrow H$ of R and for every $I \subseteq \{2,\ldots,|X|\}$, we associate the set $Con(X,I)$ of vertices of H which are connected to $\{X(i) \mid i \in I\}$ in $R^{\omega}(H)$. These sets $Con(X,I)$ are computable as being the least fixpoints of the following system:

$$\begin{aligned} Con(X,I) \ = \ & \{X(i) \mid i \in I\} \cup \bigcup \{ \ V_Y \mid Y \in H \ \wedge \ Y(1) \in T \ \wedge \ V_Y \cap Con(X,I) \ \neq \ \emptyset \ \} \ \cup \\ & \bigcup \{ \ \{Y(i) \mid Z(i) \in Con(Z, \{j | Y(j) \in Con(X,I)\})\} \mid Y \in H \\ & \wedge \ Z \in Dom(R) \ \wedge \ Y(1) = Z(1) \ \} \end{aligned}$$

for every rule $X \longrightarrow H$ of R and $I \subseteq \{2,\ldots,|X|\}$.

Let $(X,H) \ \in Dom(R)$. We define $C_X \ := \ \{ \ Con(X,i) \mid 2 \leq i \leq |X| \ \}$ the class of connected vertices in all graph of $R^{\omega}(H)$. For each class P in C_X, we associate a hyperarc X_P labelled by a new symbol $X_P(1)$ of arity $\#(P \cap V_X)$ whose set V_{X_P} of vertices is equal to $P \cap V_X$, and such that $X_P(1) \neq Y_Q(1)$ if $(X,P) \neq (Y,Q)$. Consider the following grammar:

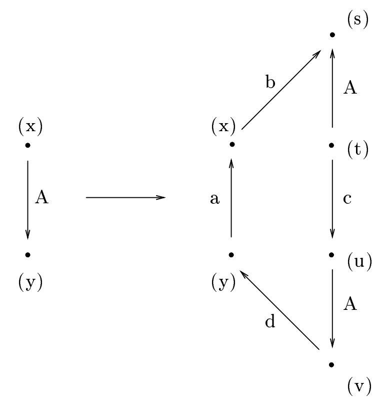
$$I \ := \ \{ \ (X, \{X_P \mid P \in C_X\}) \ \mid \ X \in Dom(R) \ \}$$

which splits each $X \in Dom(R)$ into hyperarcs according to C_X. For each rule (X,H) of R, there is a unique hypergraph H_X such that $H \underset{I}{\Longrightarrow} H_X$. In fact each part X_P of the splitting of X will generate only the part of H_X accessible from P. The *restriction* of a hypergraph H to a set V of vertices is denoted by $H_{|V} \ := \ \{ \ as_1\ldots s_p \in H \mid s_1,\ldots,s_p \in V \ \}$. As every hypergraph in $Im(R)$ does not have two non-terminal hyperarcs with the same label, the following grammar:

$$S \ := \ \{ \ (X_P, (H_X)_{|P} \mid X \in Dom(R) \ \wedge \ P \in C_X \ \}$$

generates from X_P the connected component of $R^{\omega}(X)$ containing P: $S^{\omega}(X_P) = R^{\omega}(X)_{|P}$. To generate according to S the connected components of $R^{\omega}(G)$, it suffices to take the family F of the connected components of H where $G \underset{I}{\Longrightarrow} H$, plus for each $X \in Dom(R)$ the (remaining) connected components of the H_X not containing a vertex of X. As R is reduced for every $K \in F$, $S^{\omega}(K)$ is a connected component of $R^{\omega}(G)$ and the converse is true. Note that F is not minimal: it is possible to have two hypergraphs K_1 and K_2 in F such that $S^{\omega}(K_1)$ is isomorphic to $S^{\omega}(K_2)$.
□

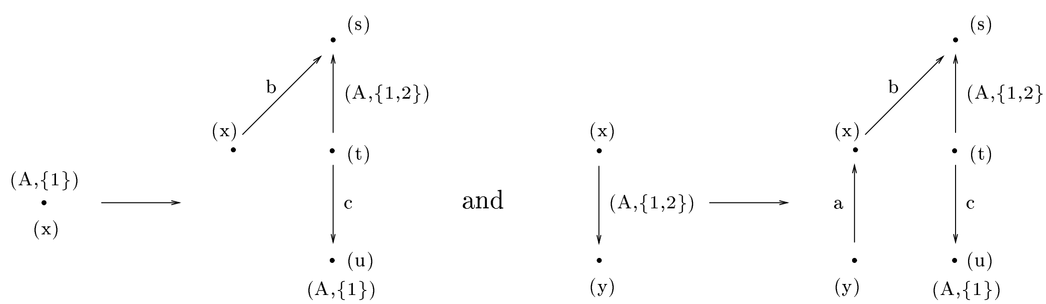
Let us apply Proposition 4.5 (a) to the following deterministic graph grammar R:



As defined in the proof of Proposition 4.5 (a), we have the following sets:

$$Acc(A, \emptyset) \ = \ \emptyset$$
$$Acc(A, \{1\}) \ = \ \{x, s, t, u\}$$
$$Acc(A, \{2\}) \ = \ Acc(A, \{1, 2\}) \ = \ \{x, y, s, t, u\} \ ,$$

and we construct the following deterministic grammar S:



such that $S^{\omega}((A, \{1\})x)$ is equal to $R^{\omega}(Axy)/x$.

Proposition 4.5 implies differents ways to have regular graphs with a root.

Corollary A.13 *The following classes of graphs are equals:*
a) *the rooted regular graphs*

b) *the rooted connected components of regular graphs*
c) *the accessible subgraphs of regular graphs.*

Proof.
a) $\Longrightarrow$ **b) :** Let G be a regular graph with a root r. Then G is the connected component of itself with a root r.
b) $\Longrightarrow$ **c) :** Let H be a connected component of a regular graph G such that H has a root r. So H is equal to the accessible subgraph of G from r.
c) $\Longrightarrow$ **a) :** Let G be a regular graph. Let H be the accessible subgraph of G from a vertex r. So r is a root of H and from Proposition 4.5 (a), H is a regular graph.
□

Corollary A.13 remains true for connected regular graphs.

Corollary A.14 *The connected regular graphs are the connected components of regular graphs.*

Proof.
Let G be a connected regular graph. Then G is the connected component of G.
Conversely let H be a connected component of a regular graph G. Then H is connected and by Proposition 4.5 (b), H is a regular graph.
□

Given an alphabet N and a finite relation R in $N^*{\times}N^*$, we denote also by

$$\underset{R}{\longmapsto} \;:=\; R.N^* \;=\; \{\ (uw,vw) \mid u\ R\ v\ \wedge\ w \in N^*\ \}$$

the *prefix rewriting* according to R, and its reflexive and transitive closure $\overset{*}{\underset{R}{\longmapsto}}$ is the *prefix derivation* according to R. A well known property is that the set $\{\ u \mid r \overset{*}{\underset{R}{\longmapsto}} u\ \}$ of words accessible by prefix derivation from a given axiom $r \in N^*$ is a rational language, and a finite automaton is exponentially constructible from R and r [Bü 64]. We give another simple construction but in an efficient way.

Proposition A.15 *We can construct in polynomial time a finite automaton recognizing the set of words obtained by prefix derivation from a given axiom and according to any rewriting system.*

Proof.
i) Let N be an alphabet and let R be a finite binary relation on N^*. We denote by

$$V \;:=\; Dom(R)\ \cup\ \{\epsilon\}$$

the set of words in the left hand sides of R, plus the empty word.
We will construct a finite graph $G\ \subseteq\ V{\times}N^*{\times}V$ (labelled on N^*) such that the language accepted by G from ϵ to any vertex u is equal to the set of words deriving by prefix from u, i.e.

$$L(G,\epsilon,\{u\}) \;=\; \{\ v \mid u \overset{*}{\underset{R}{\longmapsto}} v\ \}. \qquad (1)$$

Let $\overline{u}$ be the greatest prefix of u belonging to V, and let $\underline{u}$ be the remaining suffix, i.e.

$$\overline{u}\underline{u} = u\ \wedge\ \overline{u} \in V\ \wedge\ \forall\, v,w\ (\ (vw = \underline{u}\ \wedge\ \overline{u}v \in V)\ \Longrightarrow\ v = \epsilon\ ).$$

The prefix decomposition of the rules of R gives the following graph:

$$H \ := \ \{ \ \overline{v} \xrightarrow{v} u \mid u \ R \ v \ \wedge \ u \neq v \ \}$$

and the prefix decomposition of the vertices in V gives the following graph:

$$I \ := \ \{ \ \overline{u} \xrightarrow{ua} ua \mid ua \in V \ \wedge \ |a| = 1 \ \}.$$

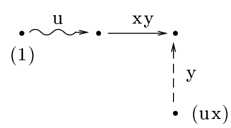
Given any graph G in $V \times N^* \times V$, a path from a vertex x to a vertex y and labelled by u is denoted by

$$x \overset{u}{\underset{G}{\Longrightarrow}} y \quad i.e. \quad u \in L(G, x, \{y\}).$$

We define a splitting $< G >$ of G as follows:

$$< G > := \ \{ \ ux \xrightarrow{y} w \mid \neg(ux \overset{y}{\Longrightarrow} w) \wedge (\exists \, v, \ \epsilon \overset{u}{\Longrightarrow} v \xrightarrow{xy} w) \wedge ux = \overline{uxy} \wedge x \neq \epsilon \ \},$$

and is represented by the following diagram:



x nonempty and maximal such that ux is in V

Note that $< G >$ is finite when G is finite.

We define a completion $\overline{G}$ of a graph G as being the least graph containing G with $< \overline{G} >= \emptyset$, i.e.

$$\begin{array}{rcll} \overline{G} & = & G & \text{if} \quad < G >= \emptyset \\ \text{and} \quad \overline{G} & = & \overline{G \cup < G >} & \text{if} \quad < G > \neq \emptyset \ . \end{array}$$

Note that $\overline{G}$ is finite when G is finite, and $\overline{G}$ can be obtained in polynomial time from G.

Let us prove that $G \ = \ \overline{H \ \cup \ I}$ satisfies property (1).

ii) Let us show that $u \overset{v}{\underset{I}{\Longrightarrow}} uv$ for every $u, uv \ \in V$.

The proof is by induction on the number

$$n_{u,v} \ = \ \#\{ \ uw \in V \mid \exists \ x \in N^*, \ wx = v \ \} \quad \geq \ 1$$

of prefixes w of v such that uw is in V.

$n_{u,v} = 1$: so $uv = u$ i.e. $v = \epsilon$. Hence $u \overset{v}{\underset{I}{\Longrightarrow}} uv$.

$n_{u,v} > 1$: so $v \neq \epsilon$. Let $ya = v$ with $|a| = 1$.

By definition of I, we have $\overline{uy} \overset{uya}{\underset{I}{\Longrightarrow}} uya$. As $n_{u,v} > 1$, there is w such that $\overline{uy} = uw$.

So $uw \overset{xa}{\underset{I}{\longrightarrow}} uv$ with $wx = y$. Furthermore $n_{u,w} = n_{u,v} - 1$.

By induction hypothesis $u \overset{w}{\underset{I}{\Longrightarrow}} uw$. As $wxa = ya = v$, we obtain $u \overset{v}{\underset{I}{\Longrightarrow}} uv$.

iii) We say that a graph J in $V \times N^* \times V$ is *full* if

$$(x \overset{u}{\underset{J}{\Longrightarrow}} y \ \wedge \ xu = zv \ \wedge \ z \in V) \quad \Longrightarrow \quad (z \overset{v}{\underset{J}{\Longrightarrow}} y).$$

Let us show that any graph J containing I and with an empty splitting $< J >$ is a full graph, i.e.

$$I \subseteq J \wedge \ < J >= \emptyset \quad \Longrightarrow \quad J \text{ is full.}$$

Let $J \ \supseteq \ I$ with $< J >= \emptyset$. Suppose that $x \overset{u}{\underset{J}{\Longrightarrow}} y$ and $xu = zv$ with $z \in V$.

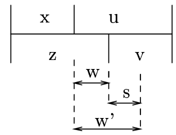
We want to show that $z \overset{v}{\underset{J}{\Longrightarrow}} y$ by distinguish the two complementary cases below.

Case 1: $|v| \geq |u|$. There is w such that $v = wu$ and $x = zw$.

As $x, z \in V$ and $x = zw$, we have by (ii) $\ z \overset{w}{\underset{I}{\Longrightarrow}} x$.

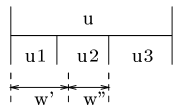
So $z \overset{w}{\underset{I}{\Longrightarrow}} x \overset{u}{\underset{J}{\Longrightarrow}} y$. As $I \ \subseteq \ J$ and $wu = v$, we have $z \overset{v}{\underset{J}{\Longrightarrow}} y$.

Case 2: $|v| < |u|$. There is w such that $u = wv$ and $z = xw$.



As $xw = z \in V$ we define w' by $xw' = \overline{xu}$. So there is s such that $ws = w'$.
As $xw', xw \in V$ and by (ii), $xw \overset{s}{\underset{I}{\Longrightarrow}} xw'$.
As $x \overset{u}{\underset{J}{\Longrightarrow}} y$ and $u \neq \epsilon$, there are $u_1, u_2, u_3 \in N^*$ and $y_1, y_2 \in V$ such that

$$x \overset{u_1}{\underset{J}{\Longrightarrow}} y_1 \overset{u_2}{\underset{J}{\Longrightarrow}} y_2 \overset{u_3}{\underset{J}{\Longrightarrow}} y \quad \wedge \quad u_1u_2u_3 = u \quad \wedge \quad |u_1| < |w'| \leq |u_1u_2|.$$



By (ii), $\epsilon \overset{x}{\underset{I}{\Longrightarrow}} x$. As $I \subseteq J$, we have $\epsilon \overset{xu_1}{\underset{J}{\Longrightarrow}} y_1$. Let w'' be such that $w'w'' = u_1u_2$.
As $xw' = \overline{xu} = \overline{xw'w''u_3}$, we obtain $xw' = \overline{xw'w''}$.
As $|u_1| < |w'|$ and $< J >= \emptyset$, we have $xw' \overset{w''}{\underset{J}{\Longrightarrow}} y_2$.
Finally $z = xw \overset{s}{\underset{I}{\Longrightarrow}} xws = xw' \overset{w''}{\underset{J}{\Longrightarrow}} y_2 \overset{u_3}{\underset{J}{\Longrightarrow}} y$.
We have $wsw''u_3 = w'w''u_3 = u_1u_2u_3 = u = wv$, hence $sw''u_3 = v$.
Having $I \subseteq J$ then $z \overset{v}{\underset{J}{\Longrightarrow}} y$.

iv) Let us prove the direct inclusion of property (1).
Let M be the set of graphs K in $V \times N^* \times V$ such that for any transition $x \overset{u}{\longrightarrow} y$ in K, $y \overset{*}{\underset{R}{\longmapsto}} xu$.
Immediately the graphs H and I are in M. Furthermore if $K \in M$ then $< K > \in M$ and $\overline{K} \in M$.
Hence $G = \overline{H \cup I}$ belongs to M. In particular if $\epsilon \overset{v}{\underset{G}{\Longrightarrow}} u$ then $u \overset{*}{\underset{R}{\longmapsto}} v$.

v) Let us prove the inverse inclusion of property (1).
Let us show by induction on $n \geq 0$ that $u \in V \wedge u \overset{n}{\underset{R}{\longmapsto}} v \quad \Longrightarrow \quad \epsilon \overset{v}{\underset{G}{\Longrightarrow}} u$.
$n = 0$: $u = v$. By (ii), $\epsilon \overset{u}{\underset{I}{\Longrightarrow}} u$ hence $\epsilon \overset{v}{\underset{G}{\Longrightarrow}} u$.
$n \Longrightarrow n+1$: $u \overset{n}{\underset{R}{\longmapsto}} w \longmapsto v$.

By induction hypothesis $\epsilon \overset{w}{\underset{G}{\Longrightarrow}} u$.
By definition of $\underset{R}{\longmapsto}$, there are $x \; R \; y$ and z such that $w = xz$ and $v = yz$.
Case 1 : $x = y$. Hence $w = v$ so $\epsilon \overset{v}{\underset{G}{\Longrightarrow}} u$.
Case 2 : $x \neq y$. By definition of H, $\overline{y} \overset{y}{\underset{H}{\longrightarrow}} x$.
By (iii), G is full. As $\epsilon \overset{w}{\underset{G}{\Longrightarrow}} u$ with $w = xz$ and $x \in V$, we have $x \overset{z}{\underset{G}{\Longrightarrow}} u$.
By (ii), $\epsilon \overset{\overline{y}}{\underset{I}{\Longrightarrow}} \overline{y}$. Thus $\epsilon \overset{\overline{y}}{\underset{I}{\Longrightarrow}} \overline{y} \overset{y}{\underset{H}{\longrightarrow}} x \overset{z}{\underset{G}{\Longrightarrow}} u$.

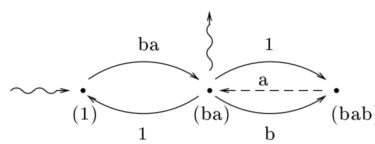
As $\overline{y}\underline{y}z = yz = v$ and $H \cup I \subseteq G$, we have $\epsilon \overset{v}{\underset{G}{\Longrightarrow}} u$.

vi) Finally and given an axiom $r \in N^*$, we construct in polynomial time from $R \cup \{(r,r)\}$ a finite automaton G on the set $Dom(R) \cup \{r,\epsilon\}$ of vertices, on the set N^* of labels, with the start state ϵ, the unique final state r, and such that G satisfies property (1).
This means that G recognizes the language $L(G,\epsilon,\{r\}) = \{ v \mid r \overset{*}{\underset{R}{\longmapsto}} v \}$ of words deriving by prefix from r and according to R.
□

Let us apply Proposition A.15 to determine the language L generated by prefix derivation from $r = ba$ according to the system $R = \{(\epsilon,ba),(bab,ba)\}$. We obtain the following graphs:

$$\begin{array}{rcl} H & = & \{ba \overset{\epsilon}{\longrightarrow} \epsilon \ , \ ba \overset{\epsilon}{\longrightarrow} bab\} \\ I & = & \{ba \overset{b}{\longrightarrow} bab \ , \ \epsilon \overset{ba}{\longrightarrow} ba\} \\ < H \cup I > & = & \{bab \overset{a}{\longrightarrow} ba\} \\ < H \cup I \cup < H \cup I >> & = & \emptyset. \end{array}$$

Thus we deduce the following finite automaton:



recognizing L. Hence $L = ba(a+ba)^*$.

From [BN 84] we can deduce a stronger result than Proposition A.15.

Proposition 4.6 *The prefix derivation of any (unlabelled) word rewriting system is effectively a rational transduction.*

Proof.
Let $R \subseteq N^* \times N^*$. We denote by $V = Dom(R) \cup \{\epsilon\}$.
i) Let us show that for every $u,v \in N^*$,

$$u \overset{*}{\underset{R}{\longmapsto}} v \iff \exists\, x,y,w \in N^* \ \exists\, z \in V \ (u = xw \ \wedge \ v = yw \ \wedge \ x \overset{*}{\underset{R}{\longmapsto}} z \overset{*}{\underset{R}{\longmapsto}} y).$$

The sufficient condition follows by the closure of $\overset{*}{\underset{R}{\longmapsto}}$ with the right concatenation.
Let us prove the necessary condition by induction on $n \geq 0$ for $u \underset{R}{\longmapsto}^n v$.
$n = 0$: $u = v$. Hence $w = u$ and $x = y = z = \epsilon$ suit.
$n \Longrightarrow n+1$: there is s such that $u \underset{R}{\longmapsto}^n s \longmapsto v$.

By induction hypothesis, there are words $x', y', w' \in N^*$ and $z' \in V$ such that $u = x'w'$, $s = y'w'$ and $x' \overset{*}{\underset{R}{\longmapsto}} z' \overset{*}{\underset{R}{\longmapsto}} y'$.

As $s \underset{R}{\longmapsto} v$, there are $p \ R \ q$ and t such that $s = pt$ and $v = qt$.

Thus $y'w' = pt$ and we distinguish the two cases below.

Case 1 : $|w'| \geq |t|$. There is h such that $w' = ht$ and $p = y'h$.

Thus $u = x'ht$ and $v = qt$ with $x'h \overset{*}{\underset{R}{\longmapsto}} z'h = p \underset{R}{\longmapsto} q$.

Case 2 : $|w'| < |t|$. There is h such that $t = hw'$ and $y' = ph$.

Thus $u = x'w'$ and $v = qhw'$ with $x' \underset{R}{\overset{*}{\longmapsto}} z' \underset{R}{\overset{+}{\longmapsto}} qh$.

ii) We denote by $u\ S\ :=\ \{\ v \mid u\ S\ v\ \}$ the image of u by a binary relation S. Let

$$S\ :=\ \bigcup\{\ (u \underset{R^{-1}}{\overset{*}{\longmapsto}})\times(u \underset{R}{\overset{*}{\longmapsto}}) \mid u \in V\ \}.$$

By Proposition A.15, S is an effective union of binary products of rational languages. By Mezei's theorem, S is an effective recognizable relation of $N^*{\times}N^*$, hence is an effective rational relation of $N^*{\times}N^*$. By (i), the prefix derivation according to R is equal to the prefix rewriting according to S, i.e.

$$\underset{R}{\overset{*}{\longmapsto}}\ =\ \underset{S}{\longmapsto}\ .$$

It follows that $\underset{R}{\overset{*}{\longmapsto}}$ is an effective rational relation : we can construct a finite transducer recognizing $\underset{R}{\overset{*}{\longmapsto}}$.

□

Let us give a simpler proof than in [Ca 90 b] that the regular graphs and the pushdown transition graphs have the same connected components.

Theorem 4.7 *We have the following properties:*

a) *The accessible pushdown transition graphs are effectively the rooted components of pushdown transition graphs, and are effectively the rooted regular graphs of finite degree;*

b) *The connected components of pushdown transition graphs are effectively the connected regular graphs of finite degree.*

Proof.

i) Let us show that any connected regular graph of finite degree is effectively a connected component of a pushdown transition graph.

Let R be a deterministic graph grammar and let G be a finite hypergraph such that $R^{\omega}(G)$ is of finite degree.

Adding a new rule, we can assume that G is restricted to a non-terminal hyperarc of R : $G \in Dom(R)$. Furthermore by Lemma A.12 then by Lemma A.11, we can suppose that R is reduced from G, and $G(2)$ is a vertex of any graph in $R^{\omega}(G)$.

We will construct a pushdown automaton P and a configuration c such that the connected component containing c of the transition graph of P is the connected component of $R^{\omega}(G)$ containing $G(2)$.

By Proposition 4.1 it suffices to construct a labelled rewriting system S on a non-terminal set N, and an axiom $s \in N^*$ such that the connected component $(S.N^*)_s$ belongs to $(R^{\omega}(G))_{G(2)}$.

Recall that V_H is the set of vertices of any hypergraph H, and that $|X|$ is the length of any word X.

We take a new alphabet $V\ =\ \{x_1, \ldots, x_m\}$ of *variables* where

$$m\ :=\ max\{\ |X| - 1 \mid X \in Dom(R)\ \}$$

is the maximum number of vertices needed by each non-terminal hyperarc.

After a possible renaming of vertices, we can assume that

$X\ =\ X(1)x_1 \ldots x_{|X|-1}$ for every $X \in Dom(R)$

$V_H\ \cap\ V_K\ =\ V_X\ \cap\ V_Y$ for every distinct rules (X,H) and (Y,K) of R.

Let us give some notations and definitions.
A set $\overline{N}$ of non-terminals is defined by

$$\overline{N} \;:=\; \overline{V} \cup \{ (x_1,\ldots,x_p) \mid 1 \le p \le m \}$$

where $\overline{V} \;:=\; V \cup \{ Y(i) \mid \exists\, H \in Im(R),\ Y \in H \ \wedge\ 1 \le i \le |Y| \}$
is the set of vertices of R.
Let $p \ge 1$. For every word $u \in \overline{N}^*$, we define

$$\begin{aligned} u < x_1,\ldots,x_p > \;&:=\; u && \text{if } u \in V \\ &:=\; u(x_1,\ldots,x_p) && \text{if } u \notin V \end{aligned}$$

the *right addition* of $(x_1,\ldots,x_p)$ to u when u is not a variable. This operation is applied also to any transition: for every non-terminal words $u, v \in \overline{N}^*$ and every terminal $a \in T$,

$$\begin{aligned} (u \xrightarrow{a} v) < x_1,\ldots,x_p > \;&:=\; u \xrightarrow{a} v && \text{if } u \notin \overline{V} \ \wedge\ v \notin \overline{V}. \\ &:=\; u < x_1,\ldots,x_p > \xrightarrow{a} v < x_1,\ldots,x_p > && \text{otherwise.} \end{aligned}$$

This definition is extended by union to any graph labelled in T.
Finally for every $v_1,\ldots,v_p \in \overline{V}$, the *substitution* $u[v_1,\ldots,v_p]$ in any word $u \in \overline{N}^*$ of the x_i by v_i is the morphism defined on every letter of $\overline{N}$ by

$$\begin{aligned} x_i[v_1,\ldots,v_p] \;&:=\; v_i && \text{for every } 1 \le i \le p \\ (x_1,\ldots,x_q)[v_1,\ldots,v_p] \;&:=\; (v_1,\ldots,v_{min(p,q)},x_{min(p,q)+1},\ldots,x_q) && \text{for every } q \ge 1 \\ r[v_1,\ldots,v_p] \;&:=\; r && \text{for another } r \in \overline{N}. \end{aligned}$$

Recall that $[H] \;:=\; \{ ast \in H \mid a \in T \}$ is the set of terminal arcs of any hypergraph H.
To every $X \in Dom(R)$, we associate a labelled rewriting system S_X on the non-terminal set $\overline{N}$ such that the prefix transition graph $S_X.\overline{N}^*$ restricted to the connected components containing the vertices of X, belongs to $R^{\omega}(X)$. These systems $S_{X(1)}$ for $X \in Dom(R)$ are computable as being the least fixpoints of the following equations:

$$\begin{aligned} S_{X(1)} \;=\;& [H] < x_1,\ldots,x_{|X|-1} > \;\cup \\ & \bigcup \{ \ (S_{Y(1)}[Y(2),\ldots,Y(|Y|)] < x_1,\ldots,x_{|Y|-1} > \ \mid Y \in H \ \wedge\ Y(1) \notin T \ \} \end{aligned}$$

for every rule $X \longrightarrow H$ of R.
As the $R^{\omega}(X)$ are regular graphs of finite degree, the systems $S_{X(1)}$ exist. Finally we take the following system:

$$S \;:=\; S_{G(1)}[G(2),\ldots,G(|G|)]$$

and the non-terminal set N:

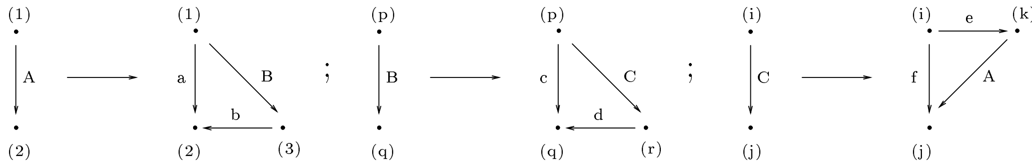
$$N \;:=\; \{ \ u(i) \mid 1 \le i \le |u| \ \wedge\ \exists\, a\ \exists\, v\ ((u \xrightarrow{a} v) \in S \ \vee\ (v \xrightarrow{a} u) \in S) \ \}$$

of the letters of S. By construction the prefix transition graph $S.N^*$ of S and the regular graph $R^{\omega}(G)$ generated by R from G have the same connected component containing $G(2)$.

ii) Let us show the converse transformation of (i) : any connected component C of any pushdown transition graph is effectively a regular graph. By Proposition A.15, the vertex set of C is an effective rational language. By Lemma A.18, the converse of (i) is a particular case of $(b) \Rightarrow (a)$ of Theorem 4.9.

iii) A particular case of (i) is that any rooted regular graph of finite degree is effectively a rooted component of a pushdown transition graph. But a rooted component of a pushdown transition graph is an accessible pushdown transition graph. Finally by Proposition 4.5 (a) and by (ii), any accessible pushdown transition graph is effectively a regular graph.
□

Let us apply proof (i) of Theorem 4.7 to the following deterministic graph grammar R:



As defined in the proof (i) of Theorem 4.7, we have the following system:

$$\begin{aligned} S_A &= \{x_1 \xrightarrow{a} x_2 \ , \ 3 \xrightarrow{b} x_2\} < x_1, x_2 > \ \cup \ S_B[x_1, 3] < x_1, x_2 > \\ S_B &= \{x_1 \xrightarrow{c} x_2 \ , \ r \xrightarrow{d} x_2\} < x_1, x_2 > \ \cup \ S_C[x_1, r] < x_1, x_2 > \\ S_C &= \{x_1 \xrightarrow{e} k \ , \ x_1 \xrightarrow{f} x_2\} < x_1, x_2 > \ \cup \ S_A[k, x_2] < x_1, x_2 > . \end{aligned}$$

This system becomes:

$$\begin{aligned} S_A &= \{x_1 \xrightarrow{a} x_2 \ , \ 3(x_1,x_2) \xrightarrow{b} x_2\} \ \cup \ S_B[x_1, 3] < x_1, x_2 > \\ S_B &= \{x_1 \xrightarrow{c} x_2 \ , \ r(x_1,x_2) \xrightarrow{d} x_2\} \ \cup \ S_C[x_1, r] < x_1, x_2 > \\ S_C &= \{x_1 \xrightarrow{e} k(x_1,x_2) \ , \ x_1 \xrightarrow{f} x_2\} \ \cup \ S_A[k, x_2] < x_1, x_2 > . \end{aligned}$$

The least fixpoint of the first component is also the least fixpoint of this unique following equation:

$$\begin{aligned} S_A = \ & \{x_1 \xrightarrow{a} x_2 \ , \ 3(x_1,x_2) \xrightarrow{b} x_2 \ , \\ & x_1 \xrightarrow{c} 3(x_1,x_2) \ , \ r(x_1,3)(x_1,x_2) \xrightarrow{d} 3(x_1,x_2) \ , \\ & x_1 \xrightarrow{e} k(x_1,r)(x_1,3)(x_1,x_2) \ , \ x_1 \xrightarrow{f} r(x_1,3)(x_1,x_2) \ \} \\ & \cup \ S_A[k,x_2] < x_1,x_2 > [x_1,r] < x_1,x_2 > [x_1,3] < x_1,x_2 > . \end{aligned}$$

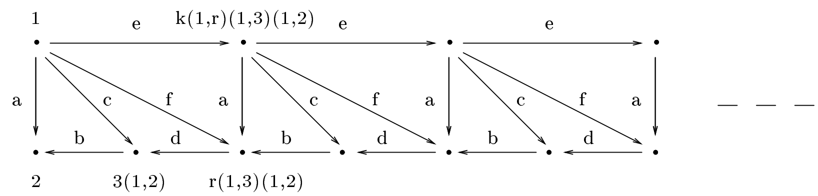
This least fixpoint is the following:

$$\begin{aligned} S_A = \ & \{x_1 \xrightarrow{a} x_2 \ , \ 3(x_1,x_2) \xrightarrow{b} x_2 \ , \\ & x_1 \xrightarrow{c} 3(x_1,x_2) \ , \ r(x_1,3)(x_1,x_2) \xrightarrow{d} 3(x_1,x_2) \ , \\ & x_1 \xrightarrow{e} k(x_1,r)(x_1,3)(x_1,x_2) \ , \ x_1 \xrightarrow{f} r(x_1,3)(x_1,x_2) \ , \\ & k(x_1,r)(x_1,3) \xrightarrow{a} r(x_1,3) \ , \ 3(k,r)(x_1,r)(x_1,3) \xrightarrow{b} r(x_1,3) \ , \\ & k(x_1,r) \xrightarrow{c} 3(k,r)(x_1,r) \ , \ r(k,3)(k,r) \xrightarrow{d} 3(k,r) \ , \\ & k(x_1,r) \xrightarrow{e} k(k,r)(k,3)(k,r)(x_1,r) \ , \ k(x_1,r) \xrightarrow{f} r(k,3)(k,r)(x_1,r) \ , \\ & k(k,r)(k,3) \xrightarrow{a} r(k,3) \ , \ 3(k,r)(k,r)(k,3) \xrightarrow{b} r(k,3) \ , \\ & k(k,r) \xrightarrow{c} 3(k,r)(k,r) \ , \\ & k(k,r) \xrightarrow{e} k(k,r)(k,3)(k,r)(k,r) \ , \ k(k,r) \xrightarrow{f} r(k,3)(k,r)(k,r) \ \}. \end{aligned}$$

Replacing x_1 by 1 and x_2 by 2, we obtain the following labelled rewriting system S:

$$\begin{array}{rclcrcl}
1 & \xrightarrow{a} & 2 & & 3(1,2) & \xrightarrow{b} & 2 \\
k(1,r)(1,3) & \xrightarrow{a} & r(1,3) & & 3(k,r)(1,r)(1,3) & \xrightarrow{b} & r(1,3) \\
k(k,r)(k,3) & \xrightarrow{a} & r(k,3) & & 3(k,r)(k,r)(k,3) & \xrightarrow{b} & r(k,3) \\
1 & \xrightarrow{c} & 3(1,2) & & r(1,3)(1,2) & \xrightarrow{d} & 3(1,2) \\
k(1,r) & \xrightarrow{c} & 3(k,r)(1,r) & & r(k,3)(k,r) & \xrightarrow{d} & 3(k,r) \\
k(k,r) & \xrightarrow{c} & 3(k,r)(k,r) & & & & \\
1 & \xrightarrow{e} & k(1,r)(1,3)(1,2) & & 1 & \xrightarrow{f} & r(1,3)(1,2) \\
k(1,r) & \xrightarrow{e} & k(k,r)(k,3)(k,r)(1,r) & & k(1,r) & \xrightarrow{f} & r(k,3)(k,r)(1,r) \\
k(k,r) & \xrightarrow{e} & k(k,r)(k,3)(k,r)(k,r) & & k(k,r) & \xrightarrow{f} & r(k,3)(k,r)(k,r)
\end{array}$$

on the set $N := \{ 1 , 2 , 3 , r , k , (1,2) , (1,3) , (1,r) , (k,3) , (k,r) \}$ of non-terminals. Then the connected component $(S.N^*)_1$ containing 1 of the prefix transition graph of S is the following graph generated by R from $A12$:



Note that the set of vertices is given directly from R by the following left linear grammar:

$$\begin{array}{rcl}
S & = & 1 \ + \ 2 \ + \ A_{1,2}(1,2) \\
A_{1,2} & = & 3 \ + \ B_{1,3}(1,3) \\
B_{1,3} & = & r \ + \ C_{1,r}(1,r) \\
C_{1,r} & = & k \ + \ A_{k,r}(k,r) \\
A_{k,r} & = & 3 \ + \ B_{k,3}(k,3) \\
B_{k,3} & = & r \ + \ C_{k,r}(k,r) \\
C_{k,r} & = & k \ + \ A_{k,r}(k,r) \ .
\end{array}$$

Hence the vertex set is the following rational language:

$$\begin{array}{l}
\{ 1 , 2 , 3(1,2) , r(1,3)(1,2) , k(1,r)(1,3)(1,2) \} \\
\cup \{ 3 , r(k,3) , k(k,r)(k,3) \}[(k,r)(k,r)(k,3)]^*(k,r)(1,r)(1,3)(1,2) \ .
\end{array}$$

Let us characterize the regular graphs of finite degree by pushdown automata with rational restrictions.

Theorem 4.8 *The following families of graphs coincide effectively:*
a) *The regular graphs of finite degree;*
b) *The rational restrictions of prefix transition graphs of labelled rewriting systems;*
c) *The rational restrictions of pushdown transition graphs.*

Proof.
c) ⇒ b) : Every pushdown transition graph is the prefix transition graph of a pushdown automaton, hence of a (labelled word) rewriting system.

b) ⇒ a) : Let R be a rewriting system on a non-terminal set N and on a terminal set T. Let $L \subseteq N^*$ be an effective rational language: we have an alphabet Q, elements $p,q \in Q$ and a rewriting system $A \subseteq Q{\times}N{\times}Q$ such that $L(A,p,\{q\}) = L$. This means that A is a nondeterministic finite automaton without ϵ-transition, with p as initial state and q as the unique final state, and A recognizes L. We may assume that the set Q of states is disjoint of N.
We will show that the restriction $(R.N^*)_{|L}$ on L of the prefix transition graph of R, is an effective regular graph.
Let \$ be a new symbol. The following rewriting system:

$$\overline{A} \ := \ \{ \ s \xrightarrow{\$} rx \mid (r \xrightarrow{x} s) \in A \ \}$$

satisfies

$$\{ \ u \mid q \underset{\overline{A}}{\longrightarrow}{}^* \ pu \ \} \ = \ L \ .$$

We group together R and $\overline{A}$ into the following rewriting system $\overline{R}$:

$$\overline{R} \ := \ \overline{A} \cup pR \ = \ \overline{A} \cup \{ \ pu \xrightarrow{a} pv \mid (u \xrightarrow{a} v) \in R \ \}$$

on the non-terminal set $N \cup Q$ and on the terminal set $T \cup \{\$\}$.
We denote by

$$K \ := \ (\overline{R}.(N \cup Q)^*)/q$$

the accessible subgraph from q of the prefix transition graph of $\overline{R}$. This graph have the two following properties:

$$K \ = \ (\overline{R}.N^*)/q$$

$$(pR.N^*)_{|L} \ = \ \{ \ u \xrightarrow{a} v \ \in K \mid a \in T \ \wedge \ \exists\, u',v' \ (u' \xrightarrow{\$} u) \in K \ \wedge \ (v' \xrightarrow{\$} v) \in K\}.$$

From Proposition 4.1 and Theorem 4.7 (a), we can construct a deterministic graph grammar S and a non-terminal hyperarc $G \in Dom(S)$ such that $K \in S^\omega(G)$. It remains to restrict the terminal arcs of any right hand side H of S to the set $Goal\$(H)$ of the vertices being a goal of a \$-transition in $S^\omega(H)$.
These sets $Goal\$(H)$ for $H \in Im(S)$ are computable as being the least fixpoints of the following equations:

$$Goal\$(H) \ := \ \{ \ v \mid \exists\, u \ (u \xrightarrow{\$} v) \in H \ \} \cup \bigcup \{ \ Y(i) \mid Y \in H \ \wedge \ 1 \le i \le |Y| \ \wedge \ \exists\, (Z,P) \in S, \ Y(1) = Z(1) \ \wedge \ Z(i) \in Goal\$(P) \ \}.$$

Finally the following deterministic graph grammar $\overline{S}$:

$$\overline{S} \ := \ \{ \ \big(X \ , \ \{ \ Y \in H \mid \exists\, Z \in Dom(R), \ Y(1) = Z(1) \ \} \cup \{ \ u \xrightarrow{a} v \mid a \in T \ \wedge \ u,v \in Goal\$(H) \ \}\big) \mid (X,H) \in S \ \}$$

generates from G the graph $(R.N^*)_{|L}$.

a) ⇒ c) : This implication is a refinement of Proposition 4.5 (c).
Let R be a deterministic graph grammar and let G be a finite hypergraph such that $R^\omega(G)$ is of finite degree.
By Lemma A.12 then by Lemma A.11, we may assume that R is proper and reduced from G. We denote by

$$N \ := \ \{ \ X(1) \mid X \in Dom(R) \ \}$$

the non-terminal set of R, and by

$$\longrightarrow \ := \ \{ \ (X(1),Y(1)) \mid \exists\, H, \ (X,H) \in R \ \wedge \ Y \in H \ \wedge \ Y(1) \in N \ \}$$

the accessibility relation on the non-terminals.
We will suppress in R the non-terminals that we need only a finite number of times to developp $R^\omega(G)$. We say that a non-terminal $x \in N$ is *useful* if there is a non-terminal

y and an infinite number of n such that $y \longrightarrow^n x$. The useful non-terminal set U is computable as being the greatest fixpoint of the following equation:

$$U \longrightarrow \;=\; U \quad \text{i.e.} \quad U \;=\; \{\, y \mid \exists\, x \in U,\; x \longrightarrow y \,\}.$$

We consider the following deterministic graph grammar S:

$$S \;:=\; \{\, (X,H) \in R \mid X(1) \notin U \,\}$$

of the restriction of R to the rules of the non-useful non-terminals.
To every hypergraph H of $Im(R-S) \cup \{G\}$, we associate a hypergraph $[H]$ such that

$$H \underset{S}{\longrightarrow}^* [H] \;\;\wedge\;\; \forall\, Y \in [H],\; Y(1) \notin U.$$

So the following deterministic graph grammar $\overline{R}$:

$$\overline{R} \;:=\; \{\, (X,[H]) \mid (X,H) \in R \;\wedge\; X(1) \in U \,\}$$

generates from $[G]$ graphs belonging to $R^\omega(G)$.
After a possible renaming (and adding new rules), we may suppose that every hypergraph in $Im(\overline{R})$ does not have two non-terminal hyperarcs with the same label. Applying the construction of Proposition 4.5 (*iii*), we define for each X in $Dom(\overline{R})$ the class C_X of connected vertices of X in $\overline{R}^\omega(X)$. Moreover for each class P in C_X, we associate a hyperarc X_P labelled by a new symbol $X_P(1)$ of arity $\#(P \cap V_X)$ whose set V_{X_P} of vertices is equal to $P \cap V_X$, and such that $X_P(1) \neq Y_Q(1)$ if $(X,P) \neq (Y,Q)$. So the following deterministic graph grammar:

$$I \;:=\; \{\, (X, \{\, X_P \mid P \in C_X \,\}) \mid X \in Dom(\overline{R}) \,\}$$

splits each $X \in Dom(\overline{R})$ into hyperarcs according to C_X .
Then the following deterministic graph grammar $\overline{S}$:

$$\overline{S} \;:=\; \{\, (X_P, H_p) \mid P \in C_X \;\wedge\; p \in P \;\wedge\; \exists\, K,\; (X,K) \in \overline{R} \;\wedge\; K \underset{I}{\Longrightarrow} H \,\}$$

generates from X_P the connected component of $\overline{R}^\omega(X)$ containing P:

$$\overline{S}^\omega(X_P) \;=\; (\overline{R}^\omega(X))_p \qquad \text{for any } p \in P.$$

Let F_1 be the following family:

$$F_1 \;:=\; \{\, H_v \mid [G] \underset{I}{\Longrightarrow} H \;\wedge\; v \in V_H \,\}$$

of the connected components of the splitting of $[G]$.
Let F_2 be the following family:

$$F_2 \;:=\; \{\, H_v \mid \exists\, X,K,\; (X,K) \in \overline{R} \wedge K \underset{I}{\Longrightarrow} H \wedge v \in V_H \wedge V_{H_v} \cap V_X \;=\; \emptyset \,\}$$

of the connected components of the splitting of any right hand side of $\overline{R}$ and not containing any vertex of its left hand side.
So $\{\, \overline{S}^\omega(H) \mid H \in F_1 \cup F_2 \,\}$ is the set of the connected components of $\overline{R}^\omega([G])$. As $\overline{R}$ has only useful non-terminals, $\overline{R}^\omega([G])$ is composed of a unique $\overline{S}^\omega(H)$ for each $H \in F_1$ and an infinite repetition of $\overline{S}^\omega(H)$ for each $H \in F_2$.
From Theorem 4.7 (b) and for each $H \in F_1 \cup F_2$, we construct a pushdown automaton P_H on a non-terminal set N_H and on a state set Q_H , and a configuration c_H such that $(P_H.(N_H)^*)_{c_H}$ belongs to $\overline{S}^\omega(H)$. We may assume that the pushdown automata P_H have distinct non-terminal sets and state sets: $N_H \cap N_K \;=\; \emptyset$ and $Q_H \cap Q_K \;=\; \emptyset$ for distinct hypergraphs H and K in $F_1 \cup F_2$. From Proposition A.15, the set $\{\, u \mid c_H \underset{P_H}{\overset{*}{\longleftrightarrow}} u \,\}$ of vertices of $(P_H.(N_H)^*)_{c_H}$ is an effective rational language L_H .
The (disjoint) union of the pushdown automata P_H gives the following pushdown automaton P:

$$P \;:=\; \bigcup\{\, P_H \mid H \in F_1 \cup F_2 \,\}$$

on the following non-terminal set $N \;:=\; \bigcup\{\, N_H \mid H \in F_1 \cup F_2 \,\}$
and on the following state set $Q \;:=\; \bigcup\{\, Q_H \mid H \in F_1 \cup F_2 \,\}$.
Let $\$$ be a new symbol and let L be the following rational language:

$$L \;:=\; \bigcup\{\, L_H \mid H \in F_1 \,\} \;\cup\; \bigcup\{\, L_H.\$^* \mid H \in F_2 \,\}.$$

Finally $(P.N^*)_{|L}$ belongs to $\overline{R}^{\omega}([G])$, i.e. to $R^{\omega}(G)$.
□

Let us give some consequences of Theorem 4.8 .

Corollary A.16 *Every pushdown transition graph is effectively a regular graph of finite degree.*

Proof.
Let P be a pushdown automaton on a non-terminal set N. The set of vertices of the transition graph $P.N^*$ of P is the following rational language:

$$L \;:=\; V_{P.N^*} \;=\; V_P.N^* \;=\; Dom(P).N^* \;\cup\; Im(P).N^*$$
$$=\; \{\, uw \mid w \in N^* \;\wedge\; \exists\, a, v\; ((u \xrightarrow{a} v) \in P \;\vee\; (v \xrightarrow{a} u) \in P) \,\}.$$

By Theorem 4.8 (c $\Rightarrow$ a), $P.N^* \;=\; (P.N^*)_{|L}$ is an effective regular graph of finite degree.
□

Furthermore Theorem 4.8 simplifies the acceptance condition on final states to test the equivalence problem of pushdown automata classes.

Corollary A.17 *For pushdown automata (respectively real-time and/or deterministic), the equivalence problem with acceptance on final states is inter-reducible to the equivalence problem with acceptance on the set of states.*

Proof.
Let P be a pushdown automaton, let $F \subseteq Q$ be a subset of states, and let $u, v \in Q.N^*$ be configurations of P.
We will construct a pushdown automaton $\overline{P}$ and two configurations $\overline{u}, \overline{v} \in \overline{Q}.\overline{N}^*$ such that

$$L(P.N^*, u, F.N^*) \;=\; L(P.N^*, v, F.N^*) \;\text{ iff }\; L(\overline{P}.\overline{N}^*, \overline{u}, \overline{Q}.\overline{N}^*) \;=\; L(\overline{P}.\overline{N}^*, \overline{v}, \overline{Q}.\overline{N}^*)\;,$$

meaning that P accepts on F the same language from u and v if and only if $\overline{P}$ accepts on its state set $\overline{Q}$ the same language from $\overline{u}$ and $\overline{v}$. We take

one new non-terminal x : $N' \;:=\; N \cup \{x\}$
three new terminals a, b, c : $T' \;:=\; T \cup \{a, b, c\}$
two new states p, q : $Q' \;:=\; Q \cup \{p, q\}$

and we construct the following pushdown automaton P':

$$P' \;:=\; P \;\cup\; \{px \xrightarrow{a} ux \;,\; px \xrightarrow{b} vx\} \;\cup\; \{\, ry \xrightarrow{c} qy \mid r \in F \;\wedge\; y \in N' \,\}.$$

We consider the set V' of configurations of P' which are accessible from px and derive to a configuration in $q.N'^*$:

$$V' \;:=\; \{\, w \in Q'.N'^* \mid px \underset{P'}{\longrightarrow}^* w \;\wedge\; \exists\, z \in N'^*,\; w \underset{P'}{\longrightarrow}^* qz \,\}$$

where $\underset{P'}{\longrightarrow} \;:=\; \{\, (sw, tw) \mid w \in N'^* \;\wedge\; \exists d\; (s \xrightarrow{d} t) \in P' \,\}$ is the unlabelled (prefix) rewriting according to P', i.e. $\underset{P'}{\longrightarrow}^*$ is the accessibility relation on the transition graph of P'. By definition of V', the restriction $(P'.N'^*)_{|V'}$ to V' of the transition graph of P', is

a locally finite graph with a root.
From Proposition 4.6, the prefix derivation $\underset{P'}{\longrightarrow}^*$ is an effective rational transduction. Consequently the set

$$V' \;=\; \{px\}\underset{P'}{\longrightarrow}^* \;\cap\; \underset{P'}{\longrightarrow}^*(q.N'^*) \;=\; \{px\}\underset{P'}{\longrightarrow}^* \;\cap\; (q.N'^*)\underset{P'^{-1}}{\longrightarrow}^*$$

is an effective rational configuration set.
From Theorem 4.8 the restriction $(P'.N'^*)_{|V'}$ to V' of the transition graph of P' is an effective regular graph. As $(P'.N'^*)_{|V'}$ is a rooted regular graph of finite degree and from Theorem4.7 (a), it is effectively an accessible pushdown transition graph. This means that we can construct a pushdown automaton $\overline{P}$ on a non-terminal set $\overline{N}$ and on a state set $\overline{Q}$, and a configuration $\overline{w}$ such that its transition graph $H \;:=\; (\overline{P}.\overline{N}^*)/\overline{w}$ accessible from $\overline{w}$ is isomorphic to $(P'.N'^*)_{|V'}$.
If P is real-time and/or deterministic then $\overline{P}$ is also respectively real-time and/or deterministic.
Let $\overline{u}$ (resp. $\overline{v}$) be the goal of the unique arc in H labelled by a (resp. b). So we have

$$\begin{array}{ll} & L(P.N^*,u,F.N^*) \;=\; L(P.N^*,v,F.N^*) \\ \text{iff} & L(P.N^*,u,F.N^*).c \;=\; L(P.N^*,v,F.N^*).c \\ \text{iff} & \text{Label}(\text{End\_Path}((P'.N'^*)_{|V'},u)) = \text{Label}(\text{End\_Path}((P'.N'^*)_{|V'},v)) \\ \text{iff} & \text{Label}(\text{End\_Path}(H,\overline{u})) = \text{Label}(\text{End\_Path}(H,\overline{v})) \\ \text{iff} & \text{Label}(\text{Path}(H,\overline{u})) = \text{Label}(\text{Path}(H,\overline{v})) \qquad \text{by Lemma A.7} \\ \text{iff} & L(\overline{P}.\overline{N}^*,\overline{u},\overline{Q}.\overline{N}^*) \;=\; L(\overline{P}.\overline{N}^*,\overline{v},\overline{Q}.\overline{N}^*). \end{array}$$

□

Let us characterize regular graphs of finite degree by rationally controlled pushdown automata. First any rational restriction of a prefix transition graph can be defined by rational control of its system.

Lemma A.18 *Given a labelled rewriting system R in $N^* \times T \times N^*$ and a rational set L of N^*, we can construct a function f from R into the family of rational sets of N^* such that $R.f \;=\; (R.N^*)_{|L}$.*

Proof.
To every rule $u \xrightarrow{a} v$ of R, we associate the rational set

$$f(u \xrightarrow{a} v) \;:=\; u^{-1}L \;\cap\; v^{-1}L$$

of words w having uw and vw in L.

$$\begin{array}{lll} \text{So} \quad (R.N^*)_{|L} & = & \{\, uw \xrightarrow{a} vw \mid u \xrightarrow{a} v \;\in R \;\wedge\; uw, vw \in L \,\} \\ & = & \{\, uw \xrightarrow{a} vw \mid u \xrightarrow{a} v \;\in R \;\wedge\; w \;\in f(u \xrightarrow{a} v) \,\} \\ & = & R.f \;. \end{array}$$

□

The converse of Lemma A.18 is false but the prefix transition graph of any rationally controlled system is isomorphic to a rational restriction of a prefix transition graph.

Theorem 4.9 *The following families of graphs coincide effectively:*
a) *The regular graphs of finite degree;*
b) *The prefix transition graphs of rationally controlled labelled rewriting systems;*
c) *The (pushdown) transition graphs of rationally controlled pushdown automata.*

Proof.
a) ⇒ c) : It follows directly from Theorem 4.8 (a) ⇒ (c) and from Lemma A.18.
c) ⇒ b) : It follows directly from the fact that any system (R,f) on (N,T) defines the same prefix transition graph $R.f$ if we extend N (and T).
b) ⇒ a) : Let R be a finite relation in $N^*{\times}T{\times}N^*$ and let f be a mapping from R into the family $Rat(N^*)$ of rational languages over N. We will construct a deterministic graph grammar S generating $R.f$ by vertices of increasing length.
Note that the set $V_{R.f}$ of vertices of $R.f$ is the following rational languages:

$$V_{R.f} \;=\; \bigcup\{\, u.f(u \xrightarrow{a} v) \mid u \xrightarrow{a} v \;\in R \,\} \cup \bigcup\{\, v.f(u \xrightarrow{a} v) \mid u \xrightarrow{a} v \;\in R \,\}.$$

We will generate $R.f$ from the empty word ϵ. So we define the following set:

$$W_{R.f} \;:=\; \{\epsilon\} \cup V_{R.f}\ .$$

Let m be the greatest length of the words in R, i.e.

$$m \;:=\; max\{\ |u| \mid \exists a\ \exists v\ (u \xrightarrow{a} v) \in R \cup R^{-1}\ \}\ .$$

To any word $u \in N^*$, we denote by s_u its suffix of length $max(0,|u|-m)$ and by p_u its corresponding prefix, i.e.

$$u = p_u s_u \quad\wedge\quad |p_u| = min(|u|,m).$$

We consider the restriction $R.f_{|u}$ of $R.f$ to the vertices of length $\geq |u|$ and having s_u as suffix. Note that $R.f_{|\epsilon} = R.f$ and $R.f_{|u} = R.f_{|v}$ if $|u| = |v| \leq m$. Note that in general $R.f_{|u}$ is not connected. We denote by

$$I_u \;:=\; \{\ v \in W_{R.f} \mid |v| = |u| \ \wedge\ s_v = s_u\ \}$$

the set of vertices of $R.f_{|u}$ of length $|u|$ (with $I_\epsilon = \{\epsilon\}$), and we denote by

$$J_u \;:=\; \{\ v \in V_{R.f} \mid \exists\ w\ \exists\ a,\ v \xrightarrow{a} w \ \in R.f \cup (R.f)^{-1} \ \wedge \\ |w| < |u| \leq |v| \ \wedge\ \exists\ z,\ s_v = z s_u\ \}$$

the set of vertices of $R.f_{|u}$ linking by an arc of $R.f$ with another vertex of length $< |u|$. In particular $J_\epsilon = \emptyset$.
We define an equivalence $\equiv$ on the set $W_{R.f}$ by $u \equiv v$ if the two following conditions are satisfied:

$$(I_u \cup J_u).s_u^{-1} \;=\; (I_v \cup J_v).s_v^{-1}$$
$$\text{and}\quad f(x \xrightarrow{a} y).s_u^{-1} \;=\; f(x \xrightarrow{a} y).s_v^{-1} \quad \forall\ x \xrightarrow{a} y\ \in R.$$

If $u \equiv v$ then $R.f_{|u}$ is isomorphic to $R.f_{|v}$: $(R.f_{|u}).s_u^{-1} = (R.f_{|v}).s_v^{-1}$.
By right quotients of any rational language, we obtain only a finite number of (rational) languages. So $\equiv$ is of finite index and a set U of representatives is constructible from (R,f).
For any $u \in U$, we associate the graph H_u of arcs of $R.f_{|u}$ with a vertex in I_u, i.e.

$$H_u \;:=\; \{\ v \xrightarrow{a} w \ \in R.f \mid |v|,|w| \geq |u| \ \wedge\ (v \in I_u \ \vee\ w \in I_u)\ \}.$$

To construct a grammar S generating $R.f$, we only add to each H_u a set K_u of non-terminal hyperarcs which generates according to S the graph $R.f_{|u}$.
To this end, we take a graded alphabet F disjoint of T, and to each $u \in U$, we associate a hyperarc j_u labelled in F such that

$$j_u = g s_1 ... s_n \ \text{ with } \ \{s_1,\ldots,s_n\} \;=\; I_u \cup J_u \quad\text{and}$$
$$s_i \neq s_j \ \text{ if } \ i \neq j \ \text{ and}$$
$$g \neq j_v(1) \ \text{ if } \ v \in U - \{u\}.$$

For any $u \in U$, we define

$$n_u \;:=\; \min(\{\ |w| \mid w \in V_{R.f_{|u}} \ \wedge\ |w| > |u|\ \} \ \cup\ \{\infty\}) \ \text{ and}$$
$$K_u \;:=\; \{\ g(t_1 s_w)\ldots(t_n s_w) \mid w \in V_{R.f_{|u}} \ \wedge\ |w| = n_u < \infty \ \wedge \\ \exists\ v \in U,\ v \ \equiv\ w \ \wedge\ j_v = g(t_1 s_v)\ldots(t_n s_v)\ \}.$$

Finally we take the following deterministic graph grammar S :

$$S \ := \ \{ \ (j_u, H_u \cup K_u) \mid u \in U \ \}.$$

Thus S is finite. For any $u \in U$, $R.f_{|u}$ belongs to $S^{\omega}(j_u)$. In particular $R.f = R.f_{|\epsilon}$ is isomorphic to $S^{\omega}(j_\epsilon)$. Hence $R.f$ is an effective regular graph of finite degree.
□

Let us apply $(b) \Rightarrow (a)$ of Theorem 4.9 to the following rationally controlled context-free grammar (P,f) :

$$P \ = \ \{ x \xrightarrow{a} xy \ , \ x \xrightarrow{b} xz \ \} \text{ on } N = \{x,y\} \text{ and } T = \{a,b\}, \text{ with}$$
$$f(x \xrightarrow{a} xy) = y^+z^* \ \text{ and } \ f(x \xrightarrow{b} xz) = z^*y^2z^*.$$

So

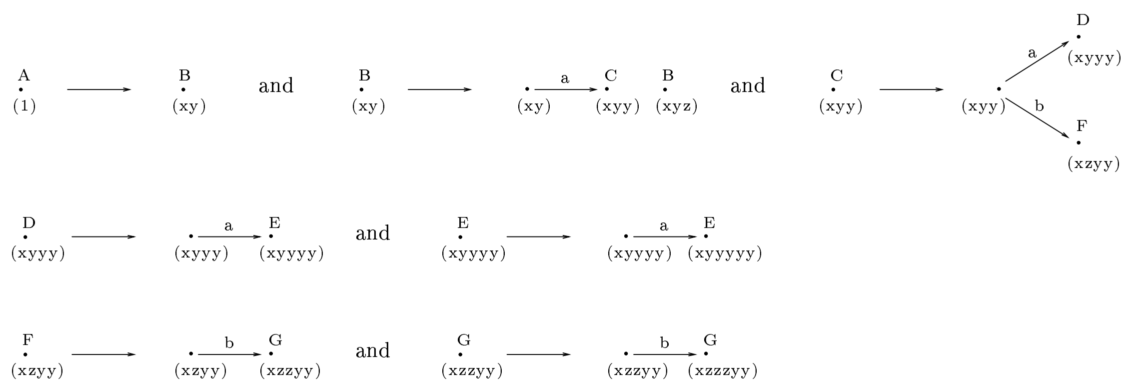
$$W_{R.f} \ = \ \epsilon \ + \ xy^+z^* \ + \ xz^*y^2z^*$$

We have the following classes according to $\equiv$:

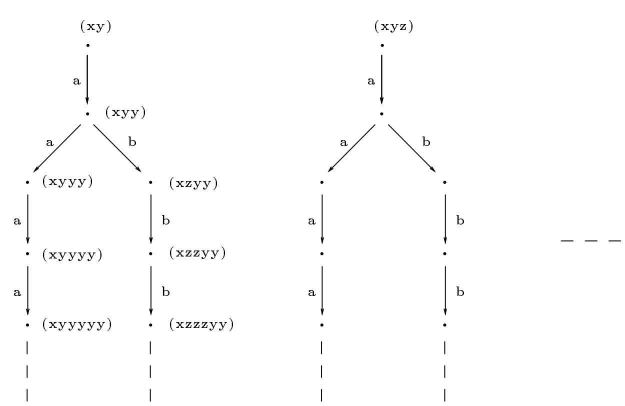
$$[\epsilon] = \{\epsilon\} \ , \ [xy] = xyz^* \ , \ [xy^2] = xy^2z^* \ , \ [xy^3] = xy^3z^* \ , \ [xy^4] = xy^4y^*z^* \ ,$$

$$[xzy^2] = xzy^2z^* \ , \ [xz^2y^2] = xz^2z^*y^2z^*.$$

We obtain the following deterministic graph grammar S :



This grammar S generates from $A\epsilon$ the following prefix transition graph $P.f$:



Let us precise the effective transformation $(b) \ \Rightarrow \ (c)$ of Theorem 4.9.

Lemma A.19 *We can transform effectively any rationally controlled rewriting system (R, f) into another system (S, g) plus an isomorphism between $R.f$ and $S.g$, and such that S is normal and ϵ-free.*

Proof.
Let R be a rewriting system in $M^* \times T \times M^*$ and let f be a mapping from R into the family $Rat(M^*)$ of rational languages over M. We will construct a rewriting system S on a set N of non-terminals, and a mapping g from S into $Rat(N^*)$ such that S is normal and ϵ-free, i.e.

$$1 \leq |u|, |v| \leq 2 \quad \text{for every } u \xrightarrow{a} v \in S,$$

and such that we have an effective isomorphism from $R.f$ to $S.g$.

i) It suffices to generalize the construction of Proposition 4.1.
We may suppose that R is *ϵ-free*. Otherwise we take a new symbol \$ and we transform (M, R, f) into another triple $(M \cup \{\$\}, \$R, f_\$)$ such that for every rule $u \xrightarrow{a} v$ of R, $f_\$(\$u \xrightarrow{a} \$v) := f(u \xrightarrow{a} v)$. Thus we have $\$R.f_\$ = \$(R.f)$.

Let m be the greatest length of the words in R, i.e.

$$m := max\{\ |u| \mid \exists a\ \exists v,\ u \xrightarrow{a} v \in R \cup R^{-1}\ \}.$$

We take an injection i from $\{u \in M^+ \mid 1 \leq |u| \leq m\}$ to some given alphabet N. We consider the injection j from M^* into N^* defined by induction as follows:

$$j(\epsilon) := \epsilon\ ,$$
$$j(u) := j(v)i(w) \qquad \text{where } u = vw \neq \epsilon \ \wedge\ |w| = min(m, |u|).$$

So the rewriting system S on N defined by

$$S := \{\ j(uw) \xrightarrow{a} j(vw) \mid (u \xrightarrow{a} v) \in R \ \wedge\ w \in M^* \ \wedge\ |w| < m\ \}$$

is normal and ϵ-free.
For every rule $x \xrightarrow{a} y$ of S, we define $g(j(x) \xrightarrow{a} j(y))$ as follows:

$$g(j(x) \xrightarrow{a} j(y)) := \bigcup\{\ j(w^{-1}.f(u \xrightarrow{a} v)) \mid u \xrightarrow{a} v \in R\ \wedge \\ |w| < m \ \wedge\ j(uw) = x \ \wedge\ j(vw) = y\ \}.$$

By extending i by inverse morphism, i.e.

$$i(u) := \{\ i(u_1)\ldots i(u_p) \mid u_1\ldots u_p = u \ \wedge\ \forall 1 \leq j \leq p,\ 1 \leq |u_j| \leq m\ \},$$

we see that j preserves the language rationality:

$$j(L) = i(L) \cap (N \cup \{\epsilon\}).\{\ A \in N \mid |i^{-1}(A)| = m\ \}^*.$$

Thus g is a mapping from S into $Rat(N^*)$.
It remains to prove that j is an isomorphism from $R.f$ to $S.g$. Note that j is a bijection from $V_{R.f}$ to $V_{S.g}$. So it remains to prove that j is a bisimulation.

ii) Let us show that j is a simulation from $R.f$ into $S.g$.
Let $u \xrightarrow{a} v$ be in $R.f$. There exist a rule $u_0 \xrightarrow{a} v_0$ of R and a word w in $f(u_0 \xrightarrow{a} v_0)$ such that $u = u_0 w$ and $v = v_0 w$. Consider the decomposition $w = xy$ where y is the greatest suffix of w such that its length $|y|$ is a multiple of m. So

$$j(u) = j(u_0 w) = j(u_0 xy) = j(u_0 x)j(y)$$
$$\text{and} \quad j(v) = j(v_0 w) = j(v_0 xy) = j(v_0 x)j(y)\ .$$

By definition of y, we have $|x| < m$. Hence $j(u_0 x) \xrightarrow{a} j(v_0 x)$ belongs to S.
As $xy = w$ we have $y \in x^{-1}.\{w\} \subseteq x^{-1}.f(u_0 \xrightarrow{a} v_0)$.
Hence $j(y) \in j(x^{-1}.f(u_0 \xrightarrow{a} v_0)) \subseteq g(j(u_0 x) \xrightarrow{a} j(v_0 x))$.
Finally $j(u) \xrightarrow{a} j(v)$ is a transition of $S.g$.

iii) Let us show that j^{-1} is a partial simulation from $S.g$ into $R.f$.
Let $j(u) \xrightarrow{a} w$ be in $S.g$. There exist a rule $p \xrightarrow{a} q$ of S and a word $t \in g(p \xrightarrow{a} q)$ such that $j(u) = pt$ and $w = qt$. By definition of g, there is a rule $u_0 \xrightarrow{a} v_0$ of R and a word w_0 of M^* such that $|w_0| < m$, $p = j(u_0 w_0)$, $q = j(v_0 w_0)$, and $t \in j(w_0^{-1}.f(u_0 \xrightarrow{a} v_0))$.
As R is ϵ-free, $p \neq \epsilon$. Furthermore $pt \in Im(j)$. So there is $x \in M^*$ such that $j(x) = t$ and $|x| = m|t|$.

$$\begin{array}{llllllll} \text{Thus} & j(u) & = & pt & = & j(u_0w_0)j(x) & = & j(u_0w_0x) \\ \text{and} & w & = & qt & = & j(v_0w_0)j(x) & = & j(v_0w_0x). \end{array}$$

As j is injective, we have $u = u_0 w_0 x$. We have $j(x) = t \in j(w_0^{-1}.f(u_0 \xrightarrow{a} v_0))$.
As j is injective, $x \in w_0^{-1}.f(u_0 \xrightarrow{a} v_0)$ hence $w_0 x \in f(u_0 \xrightarrow{a} v_0)$.
Finally $v = v_0 w_0 x$ suits: $u \xrightarrow{a} v$ is a transition of $R.f$ and $j(v) = w$.

Note that this normalization remains correct and effective if f is any mapping from R into the family $Alg(M^*)$ of algebraic languages over M : g can be constructed and is a mapping from S into $Alg(N^*)$.
□

From Lemma A.18, the class of rationally controlled prefix transition graphs is larger than the class of rationally restricted prefix transition graphs, and has basic closure properties.

Proposition 4.10 *The class of rationally controlled prefix transition graphs is closed effectively by union, difference, intersection, morphism, and rational restriction.*

Proof.
Let (R, f) and (S, g) be rationally controlled labelled rewriting systems. We may assume that R and S have the same set N of non-terminals, and the same set T of terminals : every system (R, f) on (N, T) defines the same prefix transition graph $R.f$ if we extend N and T.

i) Closure by union. We have

$$R.f \ \cup \ S.g \ = \ (R \cup S).h$$

where $h : \ R \cup S \longrightarrow Rat(N^*)$ is defined by

$$h(u \xrightarrow{a} v) \ := \ \begin{cases} f(u \xrightarrow{a} v) & \text{if } u \xrightarrow{a} v \ \in R - S \\ g(u \xrightarrow{a} v) & \text{if } u \xrightarrow{a} v \ \in S - R \\ f(u \xrightarrow{a} v) \cup g(u \xrightarrow{a} v) & \text{if } u \xrightarrow{a} v \ \in R \cap S. \end{cases}$$

ii) Closure by difference. We have

$$R.f \ - \ S.g \ = \ R.h$$

such that for every $u \xrightarrow{a} v \ \in R$, $h(u \xrightarrow{a} v)$ is defined by

$$\begin{aligned} h(u \xrightarrow{a} v) \ := \ f(u \xrightarrow{a} v) \ &- \ \bigcup\{ \ wg(uw \xrightarrow{a} vw) \mid uw \xrightarrow{a} vw \ \in S \ \} \\ &- \ \bigcup\{ \ w^{-1}g(x \xrightarrow{a} y) \mid x \xrightarrow{a} y \ \in S \ \wedge \ \exists \ w, \ xw = u \ \wedge \ yw = v \ \}. \end{aligned}$$

iii) Closure by intersection. It suffices to apply (ii) to one of the following equalities:

$$R.f \ \cap \ S.g \ = \ R.f \ - \ (R.f - S.g) \ = \ S.g \ - \ (S.g - R.f)$$

However let us establish the following direct and symmetric equality:

$$R.f \ \cap \ S.g \ = \ P.h$$

where
$P \ := \ \{\ u \xrightarrow{a} v \ \in R \mid \exists\, w,\ uw \xrightarrow{a} vw \ \in S\ \} \cup \{\ u \xrightarrow{a} v \ \in S \mid \exists\, w,\ uw \xrightarrow{a} vw \ \in R\ \}$

and $h(u \xrightarrow{a} v)$ is equal to

$$\begin{cases} f(u \xrightarrow{a} v) \ \cap \ \bigcup\{\ wg(uw \xrightarrow{a} vw) \mid uw \xrightarrow{a} vw \ \in S\ \} & \text{if } u \xrightarrow{a} v \ \in R - S \quad (1) \\ g(u \xrightarrow{a} v) \ \cap \ \bigcup\{\ wf(uw \xrightarrow{a} vw) \mid uw \xrightarrow{a} vw \ \in R\ \} & \text{if } u \xrightarrow{a} v \ \in S - R \quad (2) \\ (1) + (2) & \text{if } u \xrightarrow{a} v \ \in R \cap S. \end{cases}$$

iii-a) Let us prove that $R.f \ \cap \ S.g \ \subseteq \ P.h$.
Let $p \xrightarrow{a} q \ \in \ R.f \cap S.g$.
There are $u \xrightarrow{a} v \ \in R$ and $x \in f(u \xrightarrow{a} v)$ such that $p = ux$ and $q = vx$.
There are $s \xrightarrow{a} t \ \in S$ and $y \in g(s \xrightarrow{a} t)$ such that $p = sy$ and $q = ty$.
So $ux = sy$ and $vx = ty$. We distinguish the two complementary cases below.

Case 1: $|x| \geq |y|$. There is w such that $x = wy$.
Hence $s = uw$ and $t = vw$. This means that $uw \xrightarrow{a} vw \ \in S$, so $u \xrightarrow{a} v \ \in P$.
Furthermore $x \in f(u \xrightarrow{a} v)$ and $x \ = \ wy \ \in \ w.g(s \xrightarrow{a} t) \ \subseteq \ w.g(uw \xrightarrow{a} vw)$.
Thus $x \in h(u \xrightarrow{a} v)$.

Case 2: $|x| < |y|$. This case is similar to Case 1.

iii-b) Let us prove that $P.h \ \subseteq \ R.f \ \cap \ S.g$.
Let $p \xrightarrow{a} q \ \in \ P.h$.
There are $u \xrightarrow{a} v \ \in P$ and $x \in h(u \xrightarrow{a} v)$ such that $p = ux$ and $q = vx$.
We distinguish the two cases below.

Case 1: $u \xrightarrow{a} v \ \in R$ and $x \ \in \ f(u \xrightarrow{a} v) \ \cap \ \bigcup\{\ wg(uw \xrightarrow{a} vw) \mid uw \xrightarrow{a} vw \ \in S\ \}$.
Then there are y, z such that $uy \xrightarrow{a} vy \ \in S$, $z \in g(uy \xrightarrow{a} vy)$ and $yz = x$.
Hence $(p \xrightarrow{a} q) \ = \ (ux \xrightarrow{a} vx) \ \in \ R.f$ and $(p \xrightarrow{a} q) \ = \ (uyz \xrightarrow{a} vyz) \ \in \ S.g$.
Thus $p \xrightarrow{a} q \ \in \ R.f \cap S.g$.

Case 2: $u \xrightarrow{a} v \ \in S$ and $x \ \in \ g(u \xrightarrow{a} v) \ \cap \ \bigcup\{\ wf(uw \xrightarrow{a} vw) \mid uw \xrightarrow{a} vw \ \in R\ \}$.
This case is similar to Case 1.

iv) Closure by morphism on vertices.
Let $h :\ N^* \longrightarrow N^*$ be a morphism. We have
$$h(R.f) \ = \ h(R).g$$
such that for every $x \xrightarrow{a} y \ \in h(R),\ \ g(x \xrightarrow{a} y)$ is defined by
$$g(x \xrightarrow{a} y) \ := \ \bigcup\{\ h(f(u \xrightarrow{a} v)) \mid u \xrightarrow{a} v \ \in R \ \wedge \ h(u) = x \ \wedge \ h(v) = y\ \}.$$
Note that if R is a context-free grammar then $h(R)$ remains a context-free grammar.

v) Closure by rational restriction.
Let L be a rational language of N^*. As Lemma A.18, we have
$$(R.f)_{|L} \ = \ R.g$$
such that for every $u \xrightarrow{a} v \ \in R,\ \ g(u \xrightarrow{a} v)$ is defined by
$$g(u \xrightarrow{a} v) \ := \ f(u \xrightarrow{a} v) \ \cap \ u^{-1}L \ \cap \ v^{-1}L\ .$$
□

Note that a bounded connection mapping g of a connected graph G of finite degree has a *locally finite inverse*, i.e. $\#\{\ s \in V_G \mid g(s) = n\ \} \ <\ \infty$ for every $n \geq 0$.

Lemma A.20 *If g is a bounded connection mapping of a connected graph G of finite degree, then g^{-1} is locally finite.*

Proof.
Let us prove this lemma by contraposition.
Suppose that g is a graduation of a connected and finite degree graph G such that g^{-1} is not locally finite.
In that case, we can consider the smallest integer p such that $g^{-1}(p)$ is infinite, i.e.

$$p \ := \ min\{\ n \mid \#\{u \in V_G \mid g(u) = n\} \ = \ \infty\ \}.$$

As G is of finite degree, the subgraph

$$(G)_{g,p} \ = \ \{\ u \xrightarrow{a} v \ \in G \mid g(u) \leq p \ \wedge \ g(v) \leq p \ \wedge \ (g(u) < p \ \vee \ g(v) < p)\ \}$$

is finite. In particular the following subset E of $[G]_{g,p}$:

$$E \ := \ \{\ u \in V_{G-(G)_{g,p}} \mid g(u) = p\ \}$$

is infinite.
As G is a connected graph of finite degree and $(G)_{g,p}$ is finite, then $G - (G)_{g,p}$ has only a finite number of connected components. Hence there is an infinite subset F of E in one connected component of $G - (G)_{g,p}$. As G is of finite degree, for every $b \geq 0$ there are u, v in F such that $d_{G-(G)_{g,p}}(u,v) \ \geq \ b$. As $F \subseteq [G]_{g,p}$, this means that g is not of bounded connection.
□

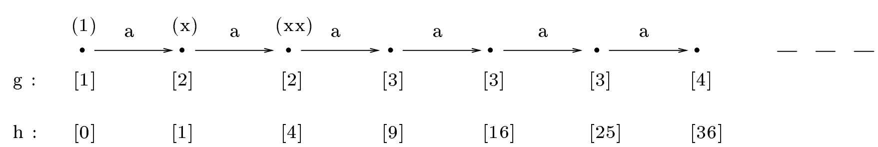
For instance, let (R, f) be the following rationally controlled system:

$$R \ = \ \{\epsilon \xrightarrow{a} x\} \quad \text{with} \quad f(\epsilon \xrightarrow{a} x) \ = \ x^*.$$

We define the following graduations g and h : for every $n \geq 0$,

$$g(x^n) \ = \ 1 + max\{\ m \mid m(m+1) \leq 2n\ \} \quad \text{and} \quad h(x^n) \ = \ n^2.$$

Thus $R.f$ is gradued by g and h as follows:



Hence g^{-1} is locally finite but g is not of bounded connection and is not vertex independant. And h^{-1} is locally finite, h is of bounded connection but h is not vertex independant.
This graph is not regular according to these graduations. Let us give a necessary and sufficient condition to preserve the regular structure of a graph using a vertex independent graduation on its vertices.

Lemma 4.12 *Given a connected rationally controlled rewriting system (R, f) and a vertex independent graduation $g : V_{R.f} \longrightarrow \mathbb{N}$, the following two properties are equivalent:*
a) *$R.f$ is a [resp. is an effective] regular graph according to g,*
b) *g is of bounded connection [resp. g^{-1} is recursive].*

Proof.

a) $\Longrightarrow$ b) : Consider a regular graph G according to a mapping g from V_G into $\mathbb{N}$. This means that there is a deterministic graph grammar S and a hyperarc $X \in Dom(S)$ such that for every $n \geq 0$, there is H with $X \underset{S}{\Longrightarrow}^n H$ and $[H] = (G)_{g,n}$ and $V_{H-[H]} = [G]_{g,n}$.
For $n = 0$, $\{ u \in V_G \mid g(u) = 0 \} \;=\; [G]_{g,0}$ hence is of cardinality $\# V_X$.
Let $n > 0$. We take one derivation $X \underset{S}{\Longrightarrow}^{n-1} K \underset{S}{\Longrightarrow} H$. We have

$$\{ u \in V_G \mid g(u) = n \} \;=\; [G]_{g,n} - [G]_{g,n-1} \;,$$

hence is of cardinality $\#(V_H - V_K)$. Thus g^{-1} is locally finite and g^{-1} is computable from S, X, g.

Let us prove that g is of bounded connection. We define the following integer:

$$b \;:=\; max\{ \; d_{S^\omega(Y)}(s,t) < \infty \mid s,t \in V_Y \;\wedge\; Y \in Dom(S) \; \}.$$

Let $n \geq 0$ and let $u, v \in [G]_{g,n}$ such that $d_{G-(G)_{g,n}}(u,v) < \infty$.
There is an hypergraph H such that $X \underset{S}{\Longrightarrow}^n H$ with $[H] = (G)_{g,n}$ and $V_{H-[H]} = [G]_{g,n}$.
As $u, v \in V_{H-[H]}$, there exists a non-terminal hyperarc $Z \in H$ with $u, v \in V_Z : u = Z(i)$ and $v = Z(j)$ for some $2 \;\leq\; i,j \;\leq\; |Z|$.
Let $Y \in Dom(S)$ such that $Y(1) = Z(1)$. Hence

$$d_{G-(G)_{g,n}}(u,v) \;=\; d_{S^\omega(Y)}(Y(i),Y(j)) \;\leq\; b \;.$$

b) $\Longrightarrow$ a) : Consider a rationally controlled rewriting system (R, f).

i) Let us verify that the accessibility $\longrightarrow^*$ in $R.f$ is decidable.
Let u and v be any vertices of $R.f$. To decide whether $u \longrightarrow^* v$, we add two new symbols \$ and & in the non-terminal set N, and two new symbols # and % in the terminal set T. We replace (R, f) by the following system (S, h) :

$$S \;:=\; \{ \; \$p \overset{a}{\longrightarrow} \$q \mid p \overset{a}{\longrightarrow} q \;\in R \; \} \;\cup\; \{ \& \overset{\%}{\longrightarrow} \$u\$ \;,\; \$v\$ \overset{\#}{\longrightarrow} \& \}$$

where $\quad h(\$p \overset{a}{\longrightarrow} \$q) \;=\; f(p \overset{a}{\longrightarrow} q).\$ \qquad$ for every $p \overset{a}{\longrightarrow} q \;\in R$

and $\quad h(\& \overset{\%}{\longrightarrow} \$u\$) \;=\; h(\$v\$ \overset{\#}{\longrightarrow} \&) \;=\; \{\epsilon\}$.

So $u \underset{R.f}{\longrightarrow}^* v \quad$ iff $\quad \& \underset{S.h}{\longrightarrow}^+ \& \quad$ iff $\quad \& \underset{S.h}{\overset{\%}{\longrightarrow}} \circ \underset{S.h}{\longrightarrow}^* \circ \underset{S.h}{\overset{\#}{\longrightarrow}} \&$.

By Theorem 4.9 and by Proposition 4.5 (a), we can construct a deterministic graph grammar U and a finite graph G generating the accessible subgraph $S.h/\&$ of $S.h$ from vertex &. Finally $u \underset{R.f}{\longrightarrow}^* v$ if and only if the label # appears in grammar U or in graph G.
This construction is simple but it would be more elegant to generalize Proposition 4.6.

ii) Let G be any graph in $V \times T \times V$.
To characterize any subgraph H of G, it suffices to know its frontier $[H]$ with $G - H$:

$$[H] \;:=\; V_H \cap V_{G-H}$$

and its subgraph $< H >$ of transitions linked to its frontier:

$$< H > := \; \{ \; u \overset{a}{\longrightarrow} v \;\in H \mid u \in V_{G-H} \;\vee\; v \in V_{G-H} \; \}.$$

In fact

$$H \;=\; G(< H >, [H])$$

where for any $K \subseteq G$ and any $E \subseteq V_K$,

$$G(K, E) \;:=\; K \cup \{ \; u \overset{a}{\longrightarrow} v \;\in G \mid \exists \, w \in V_K - E, \; w \longrightarrow^*_{/E} u \; \}$$

with $\quad u \longrightarrow_{/E} v \;$ if $\; u \longrightarrow v$ and $u, v \notin E$.

iii) Assume that $R.f$ is connected and g^{-1} is recursive.
Let m be the greatest length of the words in R, i.e.

$$m \ := \ max\{ \ |u| \mid \exists a \ \exists v \ (u \overset{a}{\longrightarrow} v) \in R \cup R^{-1} \ \} \ .$$

To any word $u \in N^*$, we denote by s_u its suffix of length $max(0, |u| - m)$. For any subgraph G of $R.f$, note that $G.s_G^{-1} \ := \ \{ \ u \overset{a}{\longrightarrow} v \mid us_G \overset{a}{\longrightarrow} vs_G \ \in G \ \}$ is isomorphic to G, where $s_G \ := \ s_u$ for any vertex u of G of minimal length.
Let $n \geq 0$. As g^{-1} is recursive and by Lemma A.20, we can compute the finite graph $(R.f)_{g,n}$.
By Proposition 4.10 and by (i), we can determine the partition $\{D_{n,1}, \ldots, D_{n,p_n}\}$ of $[R.f]_{g,n}$ into connected vertices in $R.f - (R.f)_{g,n}$.
For each $1 \leq i \leq p_n$, we can compute the subgraph

$$C_{n,i} \ := \ \{ \ u \overset{a}{\longrightarrow} v \ \in \ R.f - (R.f)_{g,n} \mid u \in D_{n,i} \ \vee \ v \in D_{n,i} \ \}.$$

Thus we have

$$R.f - (R.f)_{g,n} \ = \ \bigcup_{i=1}^{p_n} R.f(C_{n,i}, D_{n,i}).$$

Furthermore we can take a vertex $u_{n,i}$ of $R.f(C_{n,i}, D_{n,i})$ of minimal length.
Finally we can compute the maximal subgraph $H_{n,i}$ of $(R.f)_{g,n+1} - (R.f)_{g,n}$ connected to $D_{n,i}$ in $R.f - (R.f)_{g,n}$, i.e.

$$H_{n,i} \ := \ [(R.f)_{g,n+1} - (R.f)_{g,n}] \ \cap \ R.f(C_{n,i}, D_{n,i}).$$

We take an order $<$ on the non-terminal set N that we extend on N^* such that $<$ is preserved by right concatenation (for instance, by length then by lexicographic order for words of the same length). To $D_{n,i}$, we associate the hyperarc

$$U_{n,i} \ := \ (C_{n,i}.s_{u_{n,i}}^{-1}, D_{n,i}.s_{u_{n,i}}^{-1})v_1 \ldots v_q \quad \text{with} \ \{v_1, \ldots, v_q\} = D_{n,i} \text{ and } v_1 < \ldots < v_q,$$

labelled by a couple of a finite graph and of a subset of vertices.
The deterministic graph grammar S we look for is defined as the union of a sequence of grammars $(S_n)_{n \geq 0}$. This sequence is inductively constructed as follows:

$$S_0 \ := \ \emptyset$$

$$S_{n+1} \ := \ S_n \ \cup \ \{ \ (U_{n,i} \ , \ H_{n,i} \ \cup \ \bigcup\{ \ U_{n+1,j} \mid V_{U_{n+1,j}} \cap V_{H_{n,i}} \neq \emptyset \ \}) \mid 1 \leq i \leq p_n \ \wedge \ U_{n,i} \notin N(S_n) \ \}$$

where $N(S_n) \ := \ \{ \ X(1) \mid X \in Dom(S_n) \ \}$ is the non-terminal set of the graph grammar S_n.
The finitude of S is shown in (iv). We add to S a rule (X, H) such that $X(1)$ is a new symbol and $V_X = [R.f]_{g,0}$ and $\bigcup\{ \ U_{0,i} \mid 1 \leq i \leq p_0 \ \} \implies H$. As g is vertex independent, S generates $R.f$ from X and according to g.

iv) It remains to prove that S is finite, that is to say the existence of i such that $S = S_i$, i.e. $S_{i+1} = S_i$.
As $R.f$ is of finite degree and is of bounded connection, it suffices to show the existence of a bound b such that

$$\forall \ n \geq 0, \ \forall \ i, \ 1 \leq i \leq p_n, \ \exists \ u \in D_{n,i}, \ |u| - |u_{n,i}| \leq b.$$

Let p be the minimal length of the vertices of $R.f$, i.e.

$$p \ := \ min\{ \ |u| \mid u \in V_{R.f} \ \}.$$

By hypothesis $R.f$ is connected. Thus to any vertex u of $R.f$ we associate the minimal length $h(u)$ needed to access a vertex of minimal length, i.e.

$$h(u) \ := \ min\{ \ n \mid \exists \ v \in R.f, \ |v| = p \ \wedge \ u \ \longleftrightarrow_n^* \ v \ \}$$

where $u \longrightarrow_n v$ if $u \longrightarrow v$ and $|u|, |v| \leq n$.
We have seen in proof $(b) \Rightarrow (a)$ of Theorem 4.9 that $R.f$ is regular according to the length of vertices. So the following integer:

$$c \ := \ max\{\ h(u) \mid u \in V_{R.f}\ \}$$

exists. Let

$$d \ := \ max\{\ g(u) \mid u \in V_{R.f} \ \wedge \ |u| = p\ \}$$

be the maximal graduation of the vertices of minimal length.
Let $n > d$ and let $1 \leq i \leq p_n$. By definition of c, there is an undirected path $v_0, \ldots, v_q$ from $v_0 = u_{n,i}$ to a vertex v_q of minimal length p such that

$$|v_j| - |u_{n,i}| \leq c \quad \text{for every } 1 \leq j \leq q.$$

As $n > d$, we have $|u_{n,i}| > p$, hence

$$k \ := \ min\{\ j \mid v_j \notin V_{R.f(C_{n,i},D_{n,i})}\ \}$$

exists. So $k > 1$ and $v_{k-1} \in D_{n,i}$ with $|v_{k-1}| - |u_{n,i}| \leq c$. Finally

$$b \ := \ max(\{c\} \ \cup \ \{\ |u| - |u_{n,i}| \mid n \leq d \ \wedge \ 1 \leq i \leq p_n \ \wedge \ u \in D_{n,i}\ \})$$

suits, hence S is finite.
□

Let us express differently a bounded connection graduation of a rationally controlled prefix transition graph.

Lemma A.21 *Given a graduation g of a rationally controlled prefix transition graph $R.f$, the following properties are equivalent:*
a) *g is of bounded connection*
b) $\exists\, b \geq 0 \ \ \forall\, n \ \geq 0 \ \ \forall\, s,t \in [R.f]_{g,n} \ , \ \ d_{R.f-(R.f)_{g,n}}(s,t) = \infty \ \ \vee \ \ d_{R.f}(s,t) \leq b$
c) $\exists\, b \geq 0 \ \ \forall\, n \ \geq 0 \ \ \forall\, s,t \in [R.f]_{g,n} \ , \ \ d_{R.f-(R.f)_{g,n}}(s,t) = \infty \ \ \vee \ \ |s| - |t| \leq b.$

Proof.
a) $\Longrightarrow$ b) : $d_{R.f}(s,t) \ \leq \ d_{R.f-(R.f)_{g,n}}(s,t)$.
b) $\Longrightarrow$ c) : $|s| - |t| \ \leq \ m.d_{R.f}(s,t)$ where m is the greatest length of the words in R.
c) $\Longrightarrow$ a) : As shown in $(b) \Rightarrow (a)$ of Theorem 4.9, there is a deterministic graph grammar S generating from a hyperarc X the graph $R.f$ according to the length of its vertices. Assume that (c) is true. So the following integer:

$$c \ := \ max\{\ d_{S^\omega(G)-H}(s,t) < \infty \mid \exists\, Y \in Dom(S) \ \wedge \ Y \underset{S}{\Longrightarrow}^b G \ \wedge \ H \subseteq G \ \wedge \ s,t \in V_{G-H}\ \}.$$

exists and suits for (a).
□

Note that the quotient $g(G)$ of a graph G by an application g is a regular graph when G is a regular graph according to g.

Lemma A.22 *If G is a [resp. an effective] regular graph according to g then $g(G)$ is a [resp. an effective] regular graph according to the identity.*

Proof.
Suppose that G is a regular graph according to g.
There is a deterministic graph grammar S and a hyperarc X such that for every $n \geq 0$,

there is H with $X \underset{S}{\Longrightarrow}^n H$ and $[H] = (G)_{g,n}$ and $V_{H-[H]} = [G]_{g,n}$.
Let us show that $g(G)$ is a regular graph according to the identity.
It suffices for each rule (Y, H) of S to identify all vertices in H which are not in Y. We may assume that the integer 0 is not a vertex in S. Given a subset E of vertices of a graph H, the quotient $H : E$ is the identification to 0 of all vertices not in E, i.e. $H : E = h_E(H)$ where

$$h_E(s) \ := \ \begin{cases} s & \text{if } \ s \in E \\ 0 & \text{otherwise} \end{cases} \quad \text{for every } s \in V_H \ .$$

The quotient of the right hand sides of S gives the following deterministic graph grammar

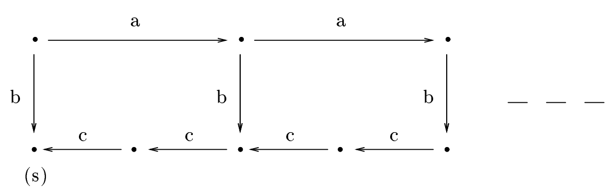
$$\overline{S} \ := \ \{ \ (Y, H : V_Y) \mid (Y, H) \in S \ \}.$$

So $\overline{S}$ generates from the hyperarc $X : \emptyset$ the graph $g(G)$ by vertices of increasing value.
□

Let us apply Lemma A.22 to the following rationally controlled system (R, f) :

$$R \ = \ \{x \xrightarrow{a} \epsilon \ , \ x \xrightarrow{b} xyy \ , \ y \xrightarrow{c} \epsilon\} \quad \text{and}$$
$$f(x \xrightarrow{a} \epsilon) \ = \ f(x \xrightarrow{b} xyy) \ = \ (yy)^* \ \text{ and } \ f(y \xrightarrow{c} \epsilon) \ = \ y^*.$$

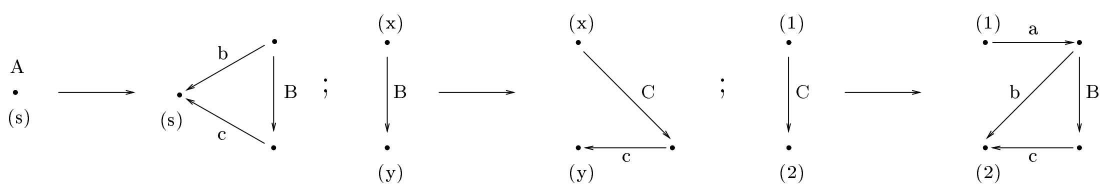
Its prefix transition graph $R.f$ is the following graph:



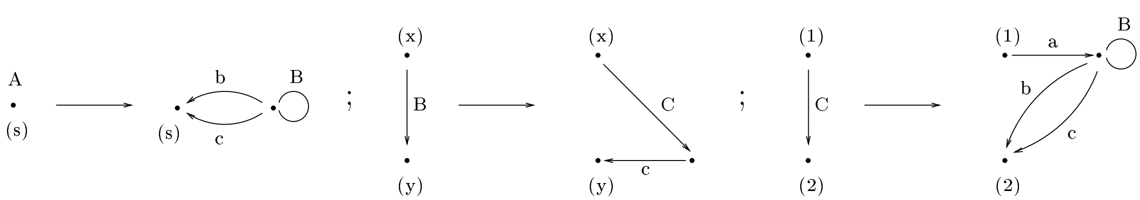
We take a graduation g on the vertives of $R.f$ as being the morphism defined by

$$g(x) \ = \ g(y) \ = \ 1 \ .$$

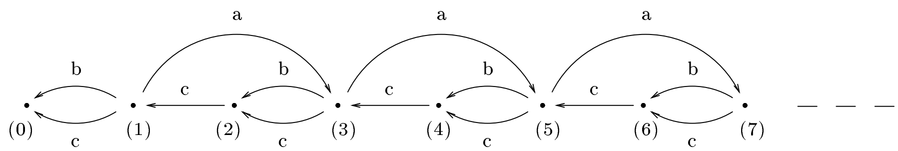
Note that g is the valuation of $R.f$: $g(u) \ = \ min\{ \ |v| \mid u \underset{R.f}{\overset{v}{\Longrightarrow}} \epsilon \ \}$ for every vertex u of $R.f$. The graph $R.f$ can be generated according to g with the following deterministic graph grammar:



By applying proof of Lemma A.22, we obtain the following deterministic graph grammar



This grammar generates from the hyperarc $A0$ the quotient $g(R.f)$ of $R.f$ by g, and is represented as follows:



By Theorem 4.10, the quotient $g(R.f)$ of a prefix transition graph $R.f$ by a morphism g remains a regular graph. Furthermore if g is ϵ-free then $R.f$ is a regular graph according to g.

Proposition 4.13 *Any rationally controlled prefix transition graph is an effective regular graph according to any ϵ-free morphism.*

Proof.
Let (R,f) be a rationally controlled rewriting system on the set N of non-terminals.
Let $h : N^* \longrightarrow \mathbb{N}$ be a morphism such that $h(x) > 0$ for every $x \in N$.
Let us show that $R.f$ is an effective regular graph according to h.
Note that h is not a vertex independent graduation. But Lemma 4.12 can be extended to h. In fact and given any vertex u of $R.f$, recall that

s_u is the suffix of u of length $max(0,|u|-m)$

where $m := max\{ \ |u| \mid \exists a \ \exists v \ (u \overset{a}{\longrightarrow} v) \in R \cup R^{-1} \ \}$

is the greatest length of the words in R,

and $((R.f))_{g,u}$ is the connected component of $(R.f)_{g,u}$ containing u.

To every vertex v of $((R.f))_{g,u}$, we associate $j(v) = v.s_u^{-1}$. So j is an isomorphism from $((R.f))_{g,u}$ to $((R.f))_{g,u}.s_u^{-1}$ satisfying

$$h(v) - h(u) = h(j(v)) - h(j(u)) \quad \text{for every vertex } v \text{ of } ((R.f))_{g,u}.$$

Thus it remains to prove that h^{-1} is recursive and that h is of bounded connection.
Let $n \geq 0$. We have

$$\{ \ u \in V_{R.f} \mid h(u) = n \ \} \ \subseteq \ \{ \ u \in V_{R.f} \mid |u| \leq n \ \},$$

hence h^{-1} is locally finite and recursive.
Consider the following integer:

$$b \ := \ max\{ \ |h(u) - h(v)| \mid u \overset{a}{\longrightarrow} v \ \in R \ \}.$$

Let $x \overset{a}{\longrightarrow} y$ be a transition of $R.f$: there is a rule $u \overset{a}{\longrightarrow} v$ of R and $w \in N^*$ such that $uw = x$ and $vw = y$. So

$$|h(x) - h(y)| \ = \ |h(uw) - h(vw)| \ = \ |h(u) - h(v)| \ \leq \ b.$$

We say that h is *of bounded difference* (see Lemma A.25).
It remains to prove that h is of bounded connection.
Let p be the greatest length of the words in $h(N)$, i.e.

$$p \ := \ max\{ \ |h(x)| \mid x \in N \ \}.$$

To any word $u \in N^*$, we denote by q_u its suffix of length $max(0,|u|-(2m-1)p)$ and by p_u its corresponding prefix, i.e.

$$u = p_u q_u \ \wedge \ |p_u| = min(|u|,(2m-1)p).$$

We have

$$R.f \ - \ (R.f)_{h,n} \ = \ \{ \ u \overset{a}{\longrightarrow} v \ \in R.f \mid h(u) > n \ \vee \ h(v) > n \ \vee \ h(u) = h(v) = n \ \}.$$

Let u be a vertex of $R.f \ - \ (R.f)_{h,n}$. If $h(u) < n$ then there is a transition $u \overset{a}{\longrightarrow} v$ of $R.f \cup (R.f)^{-1}$ such that $h(v) > n$. As $h(v) - h(u) \ \leq \ b$, we obtain

$$h(u) \;\geq\; h(v) - b \;\geq\; n+1-b.$$

But
$$\begin{aligned} b &\leq max\{\, h(u) \mid \exists\, v, \exists\, a,\; u \xrightarrow{a} v \in R \cup R^{-1} \,\} \\ &\leq max\{\, |u|p \mid \exists\, v, \exists\, a,\; u \xrightarrow{a} v \in R \cup R^{-1} \,\} \\ &= mp. \end{aligned}$$

Finally
$$h(u) \;\geq\; n+1-mp \quad \text{for every vertex } u \text{ of } R.f - (R.f)_{h,n}$$

As
$$[R.f]_{h,n} \;=\; \{\, u \in V_{R.f} \mid h(u) = n \;\vee\; \exists\, v\; \exists\, a\; (u \xrightarrow{a} v \;\in R.f \cup (R.f)^{-1} \wedge h(u) < n < h(v)\;\},$$

we have
$$h(u) \;\leq\; n \quad \text{for every } u \in [R.f]_{h,n}.$$

Let C be a connected component of $R.f \;-\; (R.f)_{h,n}$. We take a vertex u of C in $[R.f]_{h,n}$.

i) Let us show that q_u is a common suffix of the vertices of C.
If $|u| \leq (2m-1)p$ then $q_u = \epsilon$ is a suffix of any word.
If $|u| > (2m-1)p$ then
$$h(p_u) \;\geq\; |p_u| \;=\; (2m-1)p\ .$$

Let v be a vertex of C and suppose that q_u is not a suffix of C.
Hence there is a path in C from u to v passing to a vertex xq_u such that x is of length $|x| \leq m-1$. Thus
$$\begin{aligned} n+1-mp \;\leq\; h(xq_u) \;&=\; h(x) + h(q_u) \;\leq\; (m-1)p + h(q_u) \\ &\leq\; h(p_u) + h(q_u) - mp \;=\; h(u) - mp \;\leq\; n - mp\ , \end{aligned}$$

which is a contradiction. Thus q_u is a suffix of any vertex v of C.

ii) Let us show that if $q_u \neq \epsilon$ and $v \in V_C$ then $|v.q_u^{-1}| \geq m$.
Suppose that $q_u \neq \epsilon$ i.e. $|u| > (2m-1)p$ and let $v \in V_C$.
By (i), there is w such that $v = wq_u$. We want to show that $|w| \geq m$.
Recall that
$$h(v) \;\geq\; n+1-mp\ .$$

As $h(u) \leq n$, we obtain
$$h(v) \;\geq\; h(u) - mp + 1\ .$$

Hence
$$\begin{aligned} p|w| \;\geq\; h(w) \;&=\; h(v) - h(q_u) \;\geq\; h(u) - h(q_u) - mp + 1 \;=\; h(p_u) - mp + 1 \\ &\geq\; |p_u| - mp + 1 \;=\; (2m-1)p - mp + 1 \;=\; (m-1)p + 1. \end{aligned}$$

So $|w| \;\geq\; m - 1 + \frac{1}{p} \;>\; m-1$ i.e. $|w| \geq m$.

iii) Finally for every $x, y \;\in\; V_C \cap [R.f]_{h,n}$, we have by (i) :
$$d_C(x,y) \;=\; d_{C.q_u^{-1}}(x.q_u^{-1}, yq_u^{-1})\ .$$

By (ii), $C.q_u^{-1}$ is the connected component of $R.(fq_u^{-1}) - (R.(fq_u^{-1}))_{h,n-|h(q_u)|}$ containing $x.q_u^{-1}$ (or yq_u^{-1}), where fq_u^{-1} is defined for every $s \xrightarrow{a} t \;\in R$, by
$$fq_u^{-1}(s \xrightarrow{a} t) \;:=\; f(s \xrightarrow{a} t).q_u^{-1}\ .$$

As the set of right quotients of a rational language is a finite family of rational languages, there is only a finite number of fz^{-1} for $z \in N^*$.
Note that
$$\begin{aligned} n - |h(q_u)| \;&\leq\; |h(u)| - |h(q_u)| + mp - 1 \;=\; |h(p_u)| + mp - 1 \\ &\leq\; |p_u|p + mp - 1 \;\leq\; (2m-1)p^2 + mp - 1\ . \end{aligned}$$

Then $d_C(x,y) \leq b$ where b is the following integer:

$$b \ := \ max\{ \ d_G(x,y) < \infty \mid \exists \ z \in N^*, \ \exists \ n \leq (2m-1)p^2 + mp - 1, \ x,y \in [R.fz^{-1}]_{h,n} \wedge G = R.(fz^{-1}) - (R.(fz^{-1}))_{h,n} \ \}.$$

□

Let us generalize and in an effective way the following result of [MS 85] : any accessible pushdown transition graph is a regular graph according to the distance from any vertex.

Proposition 4.14 *Any connected rationally controlled prefix transition graph is an effective regular graph according to the distance from any nonempty finite set of vertices.*

Proof.
Let $R.f$ be a rationally controlled prefix transition graph and let E be a finite subset of vertices.
By Lemma A.19, we may assume that R is normal and ϵ-free : every word in R is a letter or two letters, i.e.

$$\forall \ u \xrightarrow{a} v \ \in R, \ 1 \leq |u|, |v| \leq 2.$$

By Proposition 4.13, we can construct a deterministic graph grammar S generating from a non-terminal hyperarc X the graph $R.f$ by vertices of increasing length.
So the following integer:

$$b \ := \ max\{ \ d_{S^\omega(Y)}(s,t) < \infty \mid s,t \in V_Y \ \wedge \ Y \in Dom(S) \ \}$$

exists. Let us show that $R.f$ is an effective regular graph according to the distance d from E, i.e.

$$d(u) \ := \ d(u,E) \ = \ min\{ \ d(u,v) \mid v \in E \ \}$$

Note that d^{-1} is recursive.
By Lemma 4.12, it suffices to show that d is of bounded connection. Note that

$$\begin{aligned}(R.f)_{d,n} \ &:= \ \{ \ u \xrightarrow{a} v \ \in R.f \mid d(u) \leq n \ \wedge \ d(v) \leq n \ \wedge \ (d(u) < n \ \vee \ d(v) < n) \ \} \\ &= \ \{ \ u \xrightarrow{a} v \ \in R.f \mid d(u) < n \ \vee \ d(v) < n \ \}.\end{aligned}$$

Hence

$$R.f - (R.f)_{d,n} \ = \ \{ \ u \xrightarrow{a} v \ \in R.f \mid d(u) \geq n \ \wedge \ d(v) \geq n \ \}$$

and

$$\begin{aligned}[R.f]_{d,n} \ &:= \ \{ \ u \in V_{R.f} \mid d(u) = n \ \vee \ (u \in V_{R.f-(R.f)_{d,n}} \ \wedge \ d(u) < n) \ \} \\ &= \ \{ \ u \in V_{R.f} \mid d(u) = n \ \}.\end{aligned}$$

Let $n_E \ := \ max\{ \ |u| \mid u \in E \ \}$ be the maximal length of the words in E.
let $n \geq n_E$ and let $u,v \in [R.f]_{d,n}$ such that $d_{R.f-(R.f)_{d,n}}(u,v) < \infty$, i.e. u and v are in a same connected component $((R.f))_{d,u} = ((R.f))_{d,v}$ in $R.f - (R.f)_{d,n}$.
Let $p \ := \ min\{ \ |w| \mid w \in V_{((R.f))_{d,u}} \ \}$ be the minimal length of the vertices of this connected component.
Let w be a vertex of $((R.f))_{d,u}$ of minimal length p. In particular $d(w) \geq d(u) = n$.
Consider an undirected path of minimal length from u [resp. v] to E. As R is normal and ϵ-free, this path goes through a first vertex x [resp. y] of length p.
So x,y,w are connected in $\{ \ u \xrightarrow{a} v \ \in R.f \mid |u| \geq n \ \wedge \ |v| \geq n \ \}$ hence

$$d(x,y) \leq b \quad \text{and} \quad d(x,w) \leq b.$$

Thus

$$d(u,x) + d(x) \ = \ d(u) \ \leq \ d(w) \ \leq \ d(w,x) + d(x)$$

hence

$$d(u,x) \;\leq\; d(w,x) \;\leq\; b.$$

Similarly we have $d(v,y) \;\leq\; b$. Finally

$$d(u,v) \;\leq\; d(u,x) + d(x,y) + d(y,v) \;\leq\; 3b.$$

By Lemma A.21, d is of bounded connection.

□

A convenient property is that the generation of any rationally controlled prefix transition graph according to any graduation can be made effectively with another rationally controlled rewriting system but by increasing length of its vertices.

Lemma A.23 *Given any rationally controlled rewriting system (R,f) and any mapping $g : V_{R.f} \longrightarrow \mathbb{N}$ such that $R.f$ is an effective regular graph according to g, we can construct a rationally controlled rewriting system (S,h) plus an isomorphism i from $R.f$ into $S.h$ satisfying the following property:*

$$i(\{\; u \in V_{R.f} \mid g(u) = n \;\}) \;=\; \{\; u \in V_{S.h} \mid |u| = n+1 \;\} \quad \textit{for every } n \geq 0.$$

Proof.

It suffices to take the construction (i) of Theorem 4.7 and the construction of Lemma A.18.

□

Let us verify that bisimilar vertices of a graph with a terminal coroot, have the same valuation.

Lemma A.24 *Given any graph G with a terminal coroot,*

$$\textit{if} \quad s \equiv_G t \quad \textit{then} \quad \| s \| = \| t \| \quad \textit{for every } s,t \in V_G \;.$$

Proof.

Let c be the terminal coroot of G. Let us prove this lemma by induction on $\| s \| \geq 0$.

$\| s \| = 0$: $s = c$. As $s \equiv_G t$ and s is terminal then t is terminal.

But c is the unique terminal vertex, hence $t = c = s$. In particular $\| s \| = \| t \|$.

$\| s \| > 0$: there is an arc $s \xrightarrow{a} s'$ of G such that $\| s' \| = \| s \| - 1$.

As $s \equiv_G t$, there is an arc $t \xrightarrow{a} t'$ of G such that $t' \equiv_G s'$.

By induction hypothesis $\| s' \| = \| t' \|$.

Hence $\| t \| \;\leq \| t' \| + 1 \;=\; \| s \|$.

By symetry of s and t, it follows that $\| s \| = \| t \|$.

□

Let us simplify Lemma 4.12 when the graduation is the valuation. We say that a graduation g of a graph G is *of bounded difference* if

$$\exists\, b \geq 0 \;\; \forall\, s \xrightarrow{a} t \;\in G,\; |g(s) - g(t)| \leq b.$$

Let us give another formulation.

Lemma A.25 *Given a graph G and a mapping $g : V_G \longrightarrow \mathbb{N}$, we have:*

a) *g is of bounded difference iff $\exists\, b \geq 0,\; \forall\, n \geq 0,\; \forall\, u \in [G]_{g,n},\; n - g(u) \leq b$,*

b) *if G is a regular graph according to g and G is of finite degree then g is of bounded difference.*

Proof.
i) Let us prove the necessary condition of (a).
Suppose that g is of bounded difference : there is $b \geq 0$ such that for every transition $u \xrightarrow{a} v$ of G, we have $|g(u) - g(v)| \leq b$.
Let $n \geq 0$ and let $u \in [G]_{g,n}$. We want to show that $n - g(u) \ \leq \ b$.
Either $g(u) = n$ hence $n - g(u) \ = \ 0 \ \leq \ b$.
Or $g(u) < n$ hence there is $u \xrightarrow{a} v$ in $G \cup (G)^{-1}$ such that $g(v) > n$.
As g is of bounded difference, $g(v) - g(u) \ \leq \ b$.
Hence $\ n - g(u) \ \leq \ n + b - g(v) \ < \ n + b - n \ = \ b$.

ii) Let us prove the sufficient condition of (a).
Suppose there is $b \geq 0$ such that for every $n \geq 0$ and for every $u \in [G]_{g,n}$, we have $n - g(u) \ \leq \ b$. Let us show that g is of bounded difference.
Consider any transition $u \xrightarrow{a} v$ of G. We distinguish the three below complementary cases.

Case 1: $g(u) = g(v)$. Hence $\ |g(u) - g(v)| = 0$.

Case 2: $g(u) > g(v)$.
So $v \in [G]_{g,g(u)-1}$. By hypothesis $\ (g(u) - 1) - g(v) \ \leq \ b$.
Hence $\ |g(u) - g(v)| \ = \ g(u) - g(v) \ \leq \ b + 1$.

Case 3: $g(u) < g(v)$. This case is similar to Case 2.

iii) Let us prove (b).
Suppose that G is a finite degree regular graph according to a mapping g. This means that there is a deterministic graph grammar S and a hyperarc $X \in Dom(S)$ such that for every $n \geq 0$, there is H with $X \underset{S}{\Longrightarrow}^{n} H$ and $[H] = (G)_{g,n}$ and $V_{H-[H]} = [G]_{g,n}$.
Let us show that g is of bounded difference.
As G is of finite degree, the following integer b exists:

$$b \ := \ min\{ \ n \mid \forall \ Y \in Dom(S), \ \forall \ H \ (Y \underset{S}{\Longrightarrow}^{n} H \quad \Rightarrow \quad V_{H-[H]} \cap V_Y \ = \ \emptyset) \ \}.$$

Hence we have $|g(u) - g(v)| \leq b$ for every transition $u \xrightarrow{a} v$ of G.
□

The properties of bounded difference and of bounded connection coincide for the valuation of any finite degree regular graph with a terminal coroot.

Lemma A.26 *A rationally controlled prefix transition graph with a terminal coroot is regular according to its valuation iff its valuation is of bounded difference.*

Proof.
Consider a rationally controlled prefix transition graph $R.f$ with a terminal coroot c.
By Lemma 4.12 and by Lemma A.25 (b), it suffices to prove that if the valuation of $R.f$ is of bounded difference then it is of bounded connection.
Suppose that the valuation $\| \ \ \|$ of $R.f$ is of bounded difference. So the following integer

$$b \ := \ max\{ \ \| \, u \, \| - \| \, v \, \| \mid \exists \ a, \ u \xrightarrow{a} v \ \in \ R.f \cup (R.f)^{-1} \ \}$$

exists. Then for any vertices s and t of any subgraph $G \subseteq R.f$, we have

$$\| s \| - \| t \| \ \leq \ b.d_G(s,t) \qquad (1)$$

If $d_G(s,t) = \infty$ then inequality (1) holds else we prove (1) by induction on $d_G(s,t) \geq 0$.
Let us show that $\| \ \|$ is of bounded connection.
By Proposition 4.14, we have a deterministic graph grammar S generating from a hyperarc of $Dom(S)$ the graph $R.f$ by vertices of increasing distance from c.
Let $n \geq 0$ and let C be a connected component of

$$R.f - (R.f)_{\| \|,n} \ = \ \{ \ u \xrightarrow{a} v \ \in R.f \mid \| u \| > n \ \vee \ \| v \| > n \ \vee \ \| u \| = \| v \| = n \ \}.$$

In particular $n - b \ \leq \ \| s \|$ for every vertex s of C.
Recall that the frontier $[R.f]_{\| \|,n}$ of the n-th decomposition of $R.f$ by $\| \ \|$ is the following subset of vertices:

$$[R.f]_{\| \|,n} \ = \ \{ \ u \in V_{R.f} \mid \| u \| = n \ \vee \ \exists \ v \ \exists \ a \ (u \xrightarrow{a} v \ \in R.f \cup (R.f)^{-1} \ \wedge \ \| u \| < n < \| v \| \ \}.$$

In particular $n - b \ \leq \ \| s \| \ \leq \ n$ for every s in $[R.f]_{\| \|,n}$.
Let $u, v \ \in \ V_C \cap [R.f]_{\| \|,n}$ and let $w \in V_C$ of minimal distance from c (in $R.f$). We have

$$\| u \| - \| w \| \ \leq \ n - (n-b) \ = \ b.$$

Consider a path from u to c of minimal length $\| u \|$. This path goes through a vertex x of distance $d(w,c)$ from c. By definition of x, we have

$$\| u \| - \| x \| \ \geq \ d(u,c) - d(x,c) \ = \ d(u,x).$$

Consider the following integer:

$$c \ := \ \{ \ d_{S^\omega(Y)}(s,t) < \infty \mid Y \in Dom(S) \ \wedge \ s,t \in V_Y \ \}.$$

As $d(w,c) = d(x,c)$ and by Property (1) applied to the restriction H of $R.f$ to the vertices of distance greater or equal to $d(w,c)$, we have

$$\| w \| - \| x \| \ \leq \ b.d_H(w,x) \ \leq \ b.c \ .$$

Thus

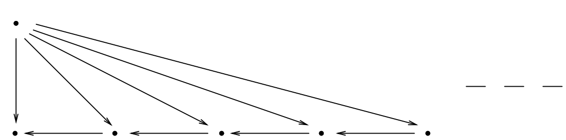
$$d(u,x) \ \leq \ \| u \| - \| x \| \ = \ (\| u \| - \| w \|) + (\| w \| - \| x \|) \ \leq \ b + bc \ = \ b(1+c)$$

hence

$$d(u,v) \ \leq \ d(u,x) + d(v,x) \ \leq \ 2b(1+c).$$

By Lemma A.21, $\| \ \|$ is of bounded connection.
□

Note that any regular graph according to the valuation is of finite in-degree but it can be of infinite out-degree. For instance, the following graph:



is regular according to the valuation and it does not satisfy Lemma A.26 : its valuation is not of bounded difference but it is of infinite out-degree.
Let us generalize the following result of [BBK 87] : the bisimulation on the prefix transition graph of any reduced context-free grammar, is decidable.

Theorem 4.16 *The bisimulation on any regular graph according to its valuation is decidable.*

Proof.

Consider a deterministic graph grammar S and a non-terminal hyperarc X such that S generates from X a graph G by vertices of increasing valuation. This means that G is a regular graph with a terminal coroot, and that we have an infinite derivation

$$X \;=\; H_0 \underset{S}{\Longrightarrow} H_1 \underset{S}{\Longrightarrow} \ldots\; H_n \underset{S}{\Longrightarrow} \ldots$$

such that for every $n \geq 0$,

$$[H_n] = (G)_{\|\ \|,n} \quad \text{and} \quad V_{H_n - [H_n]} = [G]_{\|\ \|,n}.$$

By renaming labels and by adding new rules, we may assume that every right hand side H of S have distinct non-terminals, i.e.

$$\forall\, Y,Z \;\in H - [H],\;\; Y \neq Z \;\;\Longrightarrow\;\; Y(1) \neq Z(1).$$

Furthermore and by proof of Lemma A.25 (b), we may assume that for every $m,n \geq 0$ and for every non-terminal hyperarcs $Y \in H_m - [H_m]$ and $Z \in H_n - [H_n]$ with the same label $Y(1) = Z(1)$, we have

$$\|\, Y(i)\,\| = m \quad \Longleftrightarrow \quad \|\, Z(i)\,\| = n \quad \text{for every } 2 \leq i \leq |Y|.$$

For every $n \geq 0$ and for every $Y \in H_n - [H_n]$, there is a hypergraph H_Y such that

$$Y \underset{S}{\Longrightarrow} H_Y \quad \text{and} \quad \bigcup\{\; H_Y \mid Y \;\in H_n - [H_n]\;\} \;=\; H_{n+1} - H_n\ .$$

We denote by

$$p \;:=\; max\{\; |Y| - 1 \mid Y \in Dom(S)\;\}$$

the maximal arity of S, and by

$$q \;:=\; \#S \;=\; \#\{\; Y \mid Y \in Dom(S)\;\}$$

the cardinality of S, i.e. the number of its non-terminal hyperarcs.

We take two vertices s and t of G and we want to decide whether $s \equiv_G t$.

By Lemma A.24, if $\|\, s\,\| \neq \|\, t\,\|$ then s is not bisimilar to t.

Suppose that $\|\, s\,\| = \|\, t\,\|$.

We take the following bound:

$$b \;:=\; \|\, s\,\| + q^2.2^{p^2},$$

where 2^{p^2} is a bound of the number of binary relations on vertices between two non-terminal hyperarcs.

To decide whether s and t are bisimilar, it suffices to show the following property:

$$s \equiv_G t \quad \text{iff} \quad s \equiv_{(G)_{\|\ \|,b}} t\ . \tag{1}$$

i) Let R be a bisimulation on G and let $n \geq 0$. Let us show that

$$R_{\leq n} \;:=\; \{\; (u,v) \in R \mid \|\, u\,\| \leq n \;\wedge\; \|\, v\,\| \leq n\;\}$$

is a bisimulation on $(G)_{\|\ \|,n}$.

Recall that $(G)_{\|\ \|,n} \;=\; \{\; u \overset{a}{\longrightarrow} v \;\in G \mid \|\, u\,\| \leq n \;\wedge\; \|\, v\,\| \leq n \;\wedge\; \|\, u\,\|.\|\, v\,\| < n^2\;\}$.

Consider a transition $u \overset{a}{\longrightarrow} u'$ of $(G)_{\|\ \|,n}$ and $u\ R_{\leq n}\ v$. In particular $u \overset{a}{\longrightarrow} u' \;\in G$ and $u\ R\ v$. As R is a bisimulation, there is v' such that $v \overset{a}{\longrightarrow} v' \;\in G$ and $u'\ R\ v'$. By Lemma A.24, $\|\, v\,\| = \|\, u\,\|$ and $\|\, v'\,\| = \|\, u'\,\|$. Thus $v \overset{a}{\longrightarrow} v' \;\in (G)_{\|\ \|,n}$ and $u'\ R_{\leq n}\ v'$. So $R_{\leq n}$ is a simulation. Similarly $R_{\leq n}^{-1}$ is a simulation, hence $R_{\leq n}$ is a bisimulation.

ii) Assume that $s \equiv_{(G)_{\|\ \|,b}} t$: there is a bisimulation S on $(G)_{\|\ \|,b}$ such that $s\ S\ t$.

From S, we will construct a bisimulation R on G such that $s\ R\ t$.

Let R be any binary relation on the vertices of G. For any $n \geq 0$ and for any non-terminal hyperarcs $Y,Z \in H_n - [H_n]$ of H_n, we denote by

$$R(Y,Z,n) \;:=\; \{\; ((Y(1),i),(Z(1),j)) \mid Y(i)\ R\ Z(j) \;\wedge\; \|\, Y(i)\,\| = \|\, Z(j)\,\| = n\;\}$$

the binary relation on the *places* of the vertices of Y and Z normed by n and linked by R. Furthermore for every $m \geq 0$ and for every hyperarcs $P, Q \in H_m - [H_m]$ labelled respectively by $P(1) = Y(1)$ and $Q(1) = Z(1)$, we define the following relation from the vertices of P to the vertices of Q :

$$R(Y,Z,n) < P,Q > := \{ (P(i),Q(j)) \mid (Y(1),i)\ R(Y,Z,n)\ (Z(1),j) \}$$

having the same places by R that Y with Z :

$$(R(Y,Z,n) < P,Q >)(P,Q,m) \ = \ R(Y,Z,n)$$

and

$$\| P(i) \| = \| Q(j) \| = m \quad \text{for every } (P(i),Q(j)) \in R(Y,Z,n) < P,Q >.$$

By definition of b, there is $\| s \| \leq n_0 \leq b$ such that for every $Y, Z \in H_{n_0} - [H_{n_0}]$,

$$R(Y,Z,n) \ = \ \bigcup\{ R(P,Q,m) \mid m < n \wedge P,Q \in H_m - [H_m] \wedge R(P,Q,m) \subseteq R(Y,Z,n) \}.$$

We complete the restriction

$$R_0 \ := \ S_{\leq n_0}$$

of S on $(G)_{\|,\| n_0}$ to a bisimulation $R \ := \ \bigcup\{ R_n \mid n \geq 0 \}$ on G such that for every $n \geq 0$,

$$\begin{aligned} R_{n+1} \ := \ R_n \ \cup \ \bigcup\{ & R_n(P',Q',m+1) < Y',Z' > \ | \\ & m < n_0 + n \ \wedge \ \exists\, Y,Z \ \in H_{n_0+n} - [H_{n_0+n}] \ \wedge \ \exists\, P,Q \ \in H_m - [H_m] \ , \\ & Y' \in H_Y \ \wedge \ Z' \in H_Z \ \wedge \ P' \in H_P \ \wedge \ Q' \in H_Q \\ & \wedge\, Y(1) = P(1) \ \wedge \ Z(1) = Q(1) \ \wedge \ Y'(1) = P'(1) \ \wedge \ Z'(1) = Q'(1) \\ & \wedge\, R_n(P,Q,m) \subseteq R_n(Y,Z,n_0+n) \ \} \ . \end{aligned}$$

Note that for every $n \geq 0$, we have

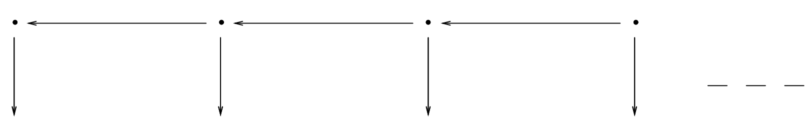
$$\| u \| = \| v \| = n_0 + n + 1 \quad \text{for every } (u,v) \in R_{n+1} - R_n \ ,$$

hence for every $0 \leq m \leq n$ and for every $Y, Z \in H_m - [H_m]$,

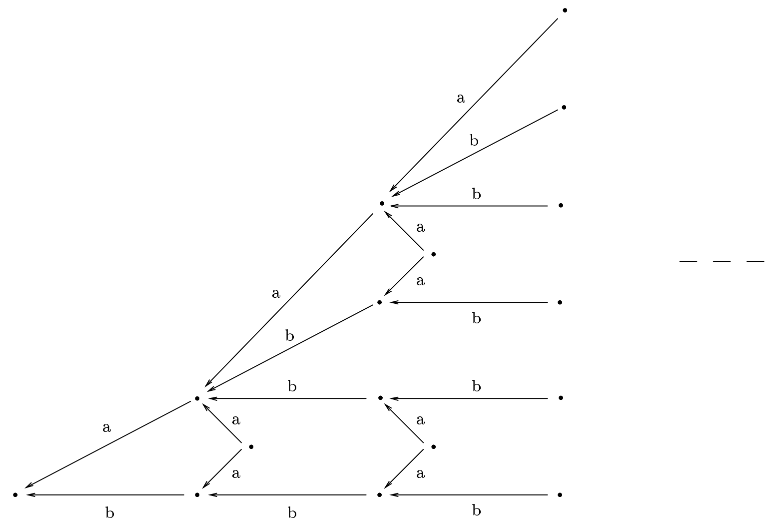
$$R_n(Y,Z,m) \ = \ R_m(Y,Z,m) \ = \ R_{max(0,m-n_0)}(Y,Z,m).$$

By induction on $n \geq 0$, R_n is a bisimulation on $(G)_{\| \ \|, n_0+n}$.
So R is a bisimulation on G.
As $n_0 \geq \| s \|$, we have $s \ R \ t$, hence $s \equiv_G t$.
☐

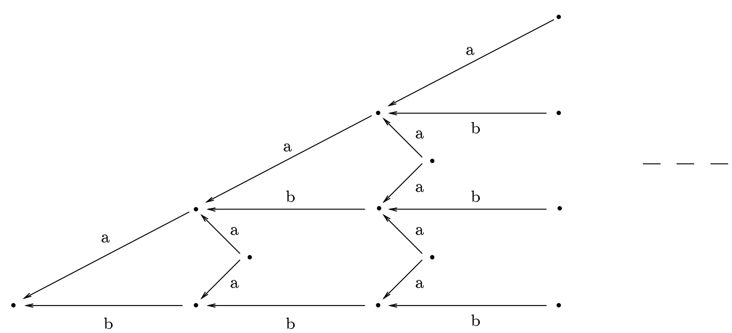
Recall that the graph following Lemma A.22 is regular according to the valuation. But the following regular graph with a terminal coroot:



is not regular according to the valuation. Furthermore the following graph:



is regular according to the valuation, but its following quotient by the greatest bisimulation:



is not a regular graph.
Nevertheless by Lemma A.24 and by Lemma A.26, for any finite degree graph which is regular according to the valuation, if its quotient by a bisimulation is regular then this quotient is also regular by valuation.

Corollary A.27 *Let G be a finite degree graph, regular according to the valuation. Let R be a bisimulation on G.*
If G/R is a regular graph then G/R is a regular graph according to the valuation.

The extension of Corollary A.27 to any regular graph of infinite (out-)degree, remains open.

A.c Term context-free grammars and regular graphs

We give some correspondences between prefix transition graphs of context-free grammars on terms, and regular graphs.
In appendix B, we study the entire rewriting systems, and show that their accessible prefix transition graphs are effectively rooted regular graphs of finite out-degree (cf Theorem B.10). The converse is true.

Theorem 5.3 *The following families of graphs coincide effectively:*
a) *The rooted regular graphs of finite out-degree;*
b) *The accessible prefix transition graphs of entire rewriting systems;*
c) *The accessible prefix transition graphs of standard and entire cf-grammars.*

Proof.
a) $\Longrightarrow$ **c) :** Let R be a deterministic graph grammar and let v be a vertex of a finite hypergraph G such that $R^{\omega}(G)$ is of finite out-degree and v is a root.
We will construct an entire and standard term cf-grammar P on a set F, and an axiom r in F_0 such that the graph $P.T(F)^*/r$ of its accessible prefix transitions from r, is isomorphic to G.
Adding a new rule, we can assume that G is restricted to a non-terminal hyperarc of R : $G \in Dom(R)$. Furthermore by Lemma A.11, we may assume that R is reduced from G.
We will construct a term context-free grammar P and an axiom such that the prefix transition graph of P accessible from its axiom belongs to $R^{\omega}(G)$.
Recall that V_H is the set of vertices of any hypergraph H, and that $|X|$ is the length of any word X.
After a possible renaming of vertices, we can assume that the rules of R have distinct vertices: $(V_X \cup V_H) \ \cap \ (V_Y \cup V_K) \ = \ \emptyset$ for every distinct rules $(X,H),(Y,K)$ of R. We gradue the set

$$F \ := \ \bigcup\{ \ V_H - V_X \mid (X,H) \in R \ \}$$

of vertices in $Im(R)$ which are not in $Dom(R)$ as follows: f is of arity $|X|-1$ for every $f \in V_H - V_X$ with $(X,H) \in R$. At the moment F_0 is empty.
We denote by T the set of terminals of R. We take a new denumerable set $\{x_1, x_2, \ldots\}$ for variables. To every vertex p of R, i.e. $p \in V_X \cup V_H$ for any rule $X \longrightarrow H$ of R, we associate a term $h(p)$ on $F \cup \{x_1, x_2, \ldots\}$ defined by

$$\begin{array}{lll} h(p) & := & x_{i-1} \quad \text{if } p = X(i) \\ h(p) & := & p x_1 \ldots x_{|X|-1} \quad \text{if } p \in V_H - V_X \end{array}$$

and we associate a labelled graph $P(p)$ such that the sets $P(p)$ are the least fixpoints of the following system:

$$P(p) \ = \ \{ \ h(p) \xrightarrow{a} h(q) \mid p \xrightarrow{a} q \in H \wedge a \in T \ \} \cup \bigcup\{ \ P(Z(i))[h(Y(2)), \ldots, h(Y(|Y|))] \mid Y \in H \ \wedge \ Y(i) = p \ \wedge \ Z \in Dom(R) \ \wedge \ Y(1) = Z(1) \ \} \ .$$

As $R^{\omega}(G)$ is a regular graph of finite out-degree, its out-degree is bounded hence the sets $P(p)$ exist. To F we add the set $F_0 = V_G$. Finally we take the following term context-free grammar:

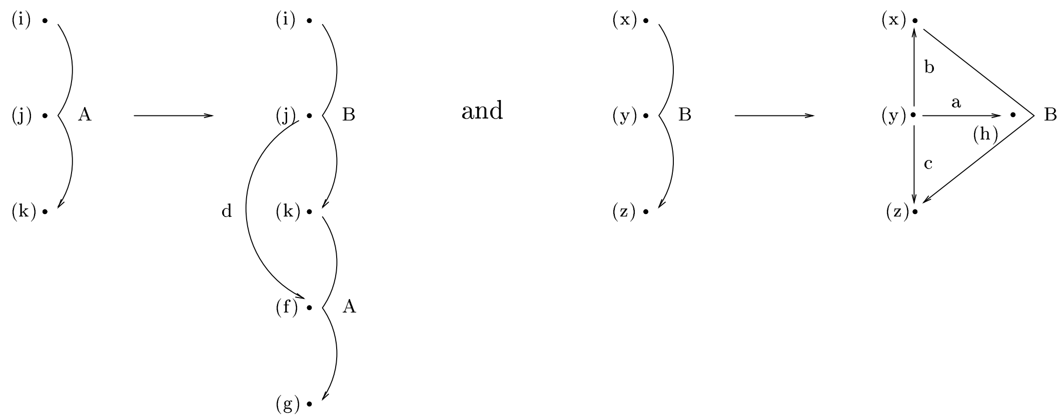
$$P \ := \ \bigcup\{ \ R(p)[G(2), \ldots, G(|G|)] \mid p \in F_0 \ \} \quad \cup \quad \bigcup\{ \ R(p) \mid p \in F - F_0 \ \}.$$

Thus P is entire and standard, and $r := v$ is in F_0.
By construction the prefix transition graph $P.T(F)^*/r$ of P and accessible from r is generated by R from G.

c) $\Longrightarrow$ b) : any term cf-grammar is a term rewriting system.

b) $\Longrightarrow$ a) : by Theorem B.10.
□

Let us apply Theorem 5.3 $(a) \Longrightarrow (c)$ to the following deterministic graph grammar R:



with axiom G reduced to $Aijk$.
As defined in the proof of Theorem 5.3, we have the following system:

$$
\begin{aligned}
P(i) &= P(x)[x_1, x_2, x_3] \\
P(j) &= \{x_2 \xrightarrow{d} fx_1x_2x_3\} \ \cup \ P(y)[x_1, x_2, x_3] \\
P(k) &= P(z)[x_1, x_2, x_3] \ \cup \ P(i)[x_3, fx_1x_2x_3, gx_1x_2x_3] \\
P(f) &= P(j)[x_3, fx_1x_2x_3, gx_1x_2x_3] \\
P(g) &= P(k)[x_3, fx_1x_2x_3, gx_1x_2x_3] \\
P(x) &= P(x)[x_1, hx_1x_2x_3, x_3] \\
P(y) &= \{x_2 \xrightarrow{a} hx_1x_2x_3 \ , \ x_2 \xrightarrow{b} x_1 \ , \ x_2 \xrightarrow{c} x_3\} \\
P(z) &= P(z)[x_1, hx_1x_2x_3, x_3] \\
P(h) &= P(y)[x_1, hx_1x_2x_3, x_3]
\end{aligned}
$$

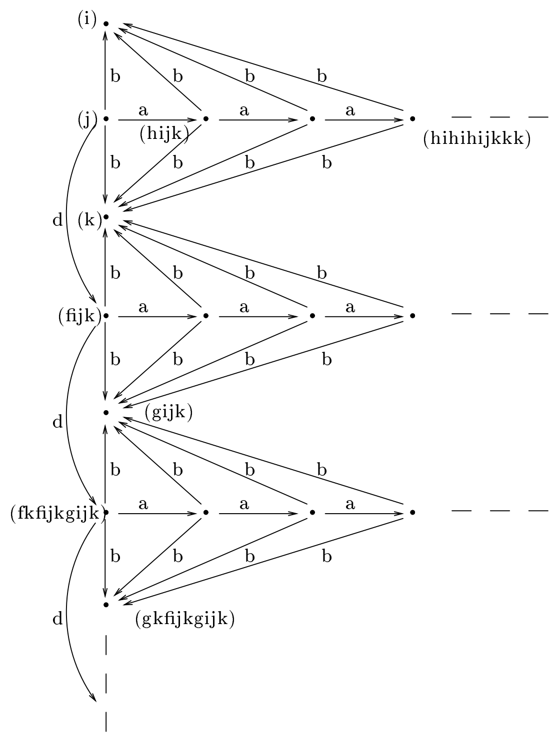
This system has the following least fixpoints:

$$
\begin{aligned}
P(i) &= P(k) \ = \ P(g) \ = \ P(x) \ = \ P(z) \ = \ \emptyset \\
P(j) &= \{x_2 \xrightarrow{a} hx_1x_2x_3 \ , \ x_2 \xrightarrow{b} x_1 \ , \ x_2 \xrightarrow{c} x_3 \ , \ x_2 \xrightarrow{d} fx_1x_2x_3\} \\
P(f) &= \{fx_1x_2x_3 \xrightarrow{a} hx_3fx_1x_2x_3gx_1x_2x_3 \ , \ fx_1x_2x_3 \xrightarrow{b} x_3 \ , \\
&\quad fx_1x_2x_3 \xrightarrow{c} gx_1x_2x_3 \ , \ fx_1x_2x_3 \xrightarrow{d} fx_3fx_1x_2x_3gx_1x_2x_3\} \\
P(y) &= \{x_2 \xrightarrow{a} hx_1x_2x_3 \ , \ x_2 \xrightarrow{b} x_1 \ , \ x_2 \xrightarrow{c} x_3\} \\
P(h) &= \{hx_1x_2x_3 \xrightarrow{a} hx_1hx_1x_2x_3x_3 \ , \ hx_1x_2x_3 \xrightarrow{b} x_1 \ , \ hx_1x_2x_3 \xrightarrow{c} x_3\}
\end{aligned}
$$

We obtain the following term context-free grammar P:

$$\begin{array}{lll} j \xrightarrow{a} hijk & fx_1x_2x_3 \xrightarrow{a} hx_3fx_1x_2x_3gx_1x_2x_3 & hx_1x_2x_3 \xrightarrow{a} hx_1hx_1x_2x_3x_3 \\ j \xrightarrow{b} i & fx_1x_2x_3 \xrightarrow{b} x_3 & hx_1x_2x_3 \xrightarrow{b} x_1 \\ j \xrightarrow{c} k & fx_1x_2x_3 \xrightarrow{c} gx_1x_2x_3 & hx_1x_2x_3 \xrightarrow{c} x_3 \\ j \xrightarrow{d} fijk & fx_1x_2x_3 \xrightarrow{d} fx_3fx_1x_2x_3gx_1x_2x_3 & \end{array}$$

Then the prefix transition graph $P.(T\{f,g,h,i,j,k\})^*/j$ of P and accessible from j is the following graph generated by G from $Aijk$:



To restrict Theorem 5.3 to regular graphs of finite degree, we consider perfect and constant-separated rewriting systems. Note that the prefix derivation of a perfect term is a perfect term if the system is perfect and constant-separated.

Lemma A.28 *Let P be a perfect and constant-separated rewriting system on a set F. Let r be a perfect term on F.*
Then its prefix transition graph $P.T(F)^*/r$ accessible from r, has only perfect vertices, and is of finite degree.

Proof.
i) Let $s \xrightarrow{a} t$ be a prefix transition of P such that s is perfect. Let us verify that t is perfect.
There is a rule $s_0 \xrightarrow{a} t_0$ in P and a substitution σ such that $s = \sigma(s_0)$ and $t = \sigma(t_0)$.
As s and s_0 are perfect, for every variable $x \in V(s_0)$, $\sigma(x)$ is perfect and of height $h(\sigma(x)) = h(s) - h(s_0)$.
Furthermore t_0 is perfect and if $V(t_0) \neq \emptyset$ then $F_0(t_0) = \emptyset$.
Case 1 : $V(t_0) = \emptyset$.

Then $t = \sigma(t_0) = t_0$ is perfect.
Case 2 : $V(t_0) \neq \emptyset$.
So $F_0(t_0) = \emptyset$.
For every perfect term r and for every substitution σ_r such that $F_0(r) = \emptyset$ and $t_0 = \sigma_r(r)$,
we have :
$\forall\ x \in V(r),\ \sigma(\sigma_r(x))$ is perfect and of height $h(\sigma(\sigma_r(x))) = (h(s) - h(s_0)) + (h(t_0) - h(r))$.
This property is proved by induction on $h(t_0) - h(r) \geq 0$.
So for $r \in V$, we have $t = \sigma(t_0) = \sigma(\sigma_r(r))$ is perfect.

ii) By (i) and by induction on the length of prefix derivations from r, any vertex of $P.T(F)^*/r$ is perfect.
For any system P, its prefix transition graph $P.T(F)^*$ is of finite out-degree.
Let $s \xrightarrow{a} t$ be any transition of $P.T(F)^*/r$.
As s and t are perfect, $h(s) \leq h(t) + max\{\ h(s_0) \mid \exists\ t_0,\ s_0 \xrightarrow{a} t_0\ \in P\ \}$.
Hence $P.T(F)^*/r$ is of finite in-degree.
□

Let us restrict Theorem 5.3 to perfect and constant-separated systems.

Theorem 5.4 *The following families of graphs coincide effectively:*
a) *The rooted regular graphs of finite degree;*
b) *The accessible prefix transition graphs of perfect, constant-separated and entire rewriting systems;*
c) *The accessible prefix transition graphs of perfect, standard and entire cf-grammars.*

Proof.
a) $\Longrightarrow$ c) : It suffices to apply proof (i) of Theorem 5.3 with Proposition 4.14.
Let us give another proof using pushdown automata.
By Theorem 4.7 (a), we can take a pushdown automaton P on a set N of non-terminals and on a set Q of states, plus an axiom $r \in QN^*$. And we will construct an entire, perfect and standard term cf-grammar P' on a set F, and an axiom r' in F_0 such that the graph $P'.T(F)^*/r'$ of its accessible prefix transitions from r', is isomorphic to the graph $P.N^*/r$ of the accessible prefix transitions of P from r.
By the proof of Proposition 4.1, we may assume that any right hand side of P has length at most 3 (i.e. belongs to $Q \cup QN \cup QNN$) and that $r \in QN$.
We denote by $\{q_1, \ldots, q_n\} := Q$ the set of n states of P.
To each couple (q, A) of a state q and of a non-terminal A, we associate a new symbol $< qA >$ of arity n. Let

$$\overline{F}\ :=\ \{\ < qA >\ \mid q \in Q\ \wedge\ A \in N\ \}.$$

For every $q \in Q$ and $U \in N^*$, we define a term $f(qU)$ as follows :

$$f(qU)\ :=\ \begin{cases} x_i & \text{if } U = \epsilon\ \wedge\ q = q_i \\ < qA > f(q_1V)\ldots f(q_nV) & \text{if } U = AV\ \wedge\ A \in N. \end{cases}$$

So the following term rewriting system :

$$f(P)\ :=\ \{\ f(s) \xrightarrow{a} f(t) \mid s \xrightarrow{a} t\ \in P\ \}$$

is a standard, perfect and entire cf-grammar without constant.
Furthermore $f(P).T(\overline{F})^*/f(r) = f(PN^*/r)$ hence is isomorphic to PN^*/r.
As $f(r)$ is not a constant, we take

$$F := \overline{F} \cup Q \cup \{r'\}$$

where each state q_i is a constant, and r' is a new symbol. And we take

$$P' := f(P) \cup \{ r' \xrightarrow{a} f(t)[q_1,\ldots,q_n] \mid r \xrightarrow{a} t \in P \}.$$

Then (P',r') suits : r' is a constant, P' is a standard, perfect and entire cf-grammar such that $P'.T(F)^*/r'$ is isomorphic to $f(P.T(F)^*/r)$ which is isomorphic to PN^*/r.

c) $\Longrightarrow$ b) : any standard term cf-grammar is a constant-separated term rewriting system.

b) $\Longrightarrow$ a) : by Theorem B.10 and by Lemma A.28.
□

An easy consequence of Theorem 5.3 is that the bisimulation decidability of prefix transition graphs of cf-grammars is inter-reducible to the bisimulation decidability of regular graphs of finite out-degree.

Proposition 5.6 *The context-free trees coincide effectively with the unfolded trees of finite out-degree regular graphs.*

Proof.
By Theorem 5.3, it suffices to show that any context-free tree is isomorphic effectively to a context-free tree of an entire cf-grammar.
Let P be a cf-grammar on $F \cup V$, and let $r \in T(F \cup V)$.
We will construct a graded alphabet $\overline{F}$, an entire cf-grammar $\overline{P}$ on $\overline{F} \cup V$, and a term $\overline{r} \in T(\overline{F} \cup V)$ such that the unfolded tree of $P.T(F)^*$ from r is isomorphic to the unfolded tree of $\overline{P}.T(\overline{F})^*$ from $\overline{r}$.
Let m be the maximal arity of the functions defined by P :

$$m := max\{ n \mid \exists\, f,\ fx_1\ldots x_n \in Dom(P) \}.$$

To each function f of F of arity n, we associate a new symbol $\overline{f}$ of arity $n+1$. Furthermore, every integer n may be seen as a function of arity n.
We consider the following graded set :

$$\overline{F} := \{ \overline{f} \mid f \in F \} \cup \{0,\ldots,m+1\} .$$

We complete each rule of P to obtain the following entire cf-grammar $\overline{P}$:

$$\overline{P} := \{\overline{f}x_1\ldots x_{n+1} \xrightarrow{a} t * (n+1) \mid fx_1\ldots x_n \xrightarrow{a} t \in P \}$$

where for every $n \geq 0$, we have

$$x * n := x \quad \text{for every variable } x$$
$$ft_1\ldots t_m * n := \overline{f}(t_1 * n)\ldots(t_m * n)n(x_1,\ldots,x_n) \quad .$$

By construction, $\overline{P}$ and $\overline{r} := r * 0$ suit.
□

For instance, let us apply the construction of Proposition 5.6 to the following cf-grammar $P = \{fxy \xrightarrow{a} fxgy\ ,\ fxy \xrightarrow{b} x\}$ with the term $r = fxy$, where f is of arity 2 and g is of arity 1.
We obtain the following entire cf-grammar $\overline{P} = \{\overline{f}xyz \xrightarrow{a} \overline{f}x\overline{g}y3xyz3xyz\ ,\ \overline{f}xyz \xrightarrow{b} x\ \}$

and the term $\overline{r} = \overline{f}xy0$. For this example, $P.T(F)^*/r$ is isomorphic to $\overline{P}.T(\overline{F})^*/\overline{r}$, hence they have the same unfolded tree.

We consider now the equivalence problem for dpda. The acceptance condition is either by empty stack, or by final states, or by all the states.

Lemma A.29 *The equivalence problem for dpda is the same with acceptance*
a) *on empty stack,*
b) *by final states,*
c) *by all the states.*

Proof.
b) $\Longleftrightarrow$ c) : By Corollary A.17.

a) $\Longrightarrow$ b) : Let P be a dpda and let $u, v \in Q.N^*$ be configurations of P.
We will construct a dpda $\overline{P}$, two configurations $\overline{u}, \overline{v} \in \overline{Q}.\overline{N}^*$, and a subset $\overline{F}$ of $\overline{Q}$ such that

$$L(P.N^*, u, Q) \;=\; L(P.N^*, v, Q) \quad \text{iff} \quad L(\overline{P}.\overline{N}^*, \overline{u}, \overline{F}.\overline{N}^*) \;=\; L(\overline{P}.\overline{N}^*, \overline{v}, \overline{F}.\overline{N}^*)\ ,$$

meaning that P accepts on empty stack the same language from u and v if and only if $\overline{P}$ accepts on $\overline{F}$ the same language from $\overline{u}$ and $\overline{v}$. We take

$$\begin{aligned} \text{one new non-terminal } \& \quad &: \quad \overline{N} \;:=\; N \cup \{\&\} \\ \text{no new terminal} \quad &: \quad \overline{T} \;:=\; T \\ \text{one new state } \# \quad &: \quad \overline{Q} \;:=\; Q \cup \{\#\} \end{aligned}$$

and we construct the following pushdown automaton $\overline{P}$:

$$\overline{P} \;:=\; P \;\cup\; \{\; p\& \xrightarrow{\epsilon} \# \mid p \in Q \;\}.$$

Thus for every non-terminal word $u \in N^*$, we have

$$L(P.N^*, u, Q)\$ \;=\; L(\overline{P}.\overline{N}^*, u\&, \{\#\}) \;=\; L(\overline{P}.\overline{N}^*, u\&, \#.\overline{N}^*)\ .$$

It remains to take $\overline{F} := \{\#\}$, $\overline{u} := u\&$ and $\overline{v} := v\&$:

$$\begin{aligned} & L(P.N^*, u, Q) \;=\; L(P.N^*, v, Q) \\ \text{iff} \quad & L(\overline{P}.\overline{N}^*, \overline{u}, \overline{F}.\overline{N}^*) \;=\; L(\overline{P}.\overline{N}^*, \overline{v}, \overline{F}.\overline{N}^*). \end{aligned}$$

b) $\Longrightarrow$ a) : Let P be a dpda, let $F \subseteq Q$ be a subset of states, and let $u, v \in Q.N^*$ be configurations of P.
We will construct a dpda $\overline{P}$ and two configurations $\overline{u}, \overline{v} \in \overline{Q}.\overline{N}^*$ such that

$$L(P.N^*, u, F.N^*) \;=\; L(P.N^*, v, F.N^*) \quad \text{iff} \quad L(\overline{P}.\overline{N}^*, \overline{u}, \overline{Q}) \;=\; L(\overline{P}.\overline{N}^*, \overline{v}, \overline{Q})\ ,$$

meaning that P accepts on F the same language from u and v if and only if $\overline{P}$ accepts on empty stack the same language from $\overline{u}$ and $\overline{v}$. We take

$$\begin{aligned} \text{one new non-terminal } \& \quad &: \quad \overline{N} \;:=\; N \cup \{\&\} \\ \text{one new terminal } \$ \quad &: \quad \overline{T} \;:=\; T \cup \{\$\} \\ \text{one new state } \# \text{ plus } |Q| \text{ new states} \quad &: \quad \overline{Q} \;:=\; Q \cup \{\#\} \cup \{\; \overline{p} \mid p \in Q \;\} \end{aligned}$$

and we construct the following pushdown automaton $\overline{P}$:

$$\begin{aligned} \overline{P} \;:=\; & \{\; pA \xrightarrow{a} qU \;\in P \mid p \notin F \vee a \neq \epsilon \;\} \cup \{\; pA \xrightarrow{\epsilon} \overline{q}U \mid pA \xrightarrow{\epsilon} qU \;\in P \wedge p \in F \;\} \\ & \cup \{\; \overline{p}A \xrightarrow{\epsilon} \overline{q}U \mid pA \xrightarrow{\epsilon} qU \;\in P \;\} \cup \{\; \overline{p}A \xrightarrow{a} qU \mid pA \xrightarrow{a} qU \;\in P \wedge a \neq \epsilon \;\} \\ & \cup \{\; \overline{p}A \xrightarrow{\$} \# \mid A \in \overline{N} \;\wedge\; \neg\, (pA \xrightarrow{\epsilon}\,) \;\} \;\cup\; \{\; p\& \xrightarrow{\$} \# \mid p \in F \;\} \\ & \cup \{\; \#A \xrightarrow{\epsilon} \# \mid A \in \overline{N} \;\}. \end{aligned}$$

Thus for every non-terminal word $u \in N^*$, we have

$$L(P.N^*, u, F.N^*)\$ \;=\; L(\overline{P}.\overline{N}^*, u\&, \{\#\}) \;=\; L(\overline{P}.\overline{N}^*, u\&, \overline{Q})\ .$$

It remains to take $\overline{u} := u\&$ and $\overline{v} := v\&$:

$$\begin{array}{lrcl} & L(P.N^*, u, F.N^*) & = & L(P.N^*, v, F.N^*) \\ \text{iff} & L(P.N^*, u, F.N^*).\$ & = & L(P.N^*, v, F.N^*).\$ \\ \text{iff} & L(\overline{P}.\overline{N}^*, \overline{u}, \overline{Q}) & = & L(\overline{P}.\overline{N}^*, \overline{v}, \overline{Q}). \end{array}$$

$\square$

Let us translate this equivalence problem for dpda to the decidability of bisimulation on regular graphs of finite out-degree.

Proposition 5.9 *The following problems are inter-reducible :*
a) *The equivalence for dpda*
b) *The bisimulation on deterministic regular graphs*
c) *The equality of deterministic context-free trees.*

Proof.
b) $\Longleftrightarrow$ c) : By Proposition 5.6.

a) $\Longrightarrow$ b) : Let P be a dpda and $u, v \in Q.N^*$ be configurations.
We will construct a deterministic regular graph H of finite out-degree, with vertices s, t such that

$$L(P.N^*, u, Q.N^*) \;=\; Label(Path(H, s)) \;\text{ and }\; L(P.N^*, v, Q.N^*) \;=\; Label(Path(H, t)).$$

By Lemma A.6, it follows the following property (1) :

$$L(P.N^*, u, Q.N^*) \;=\; L(P.N^*, v, Q.N^*) \;\text{ iff }\; s \equiv_H t \,, \qquad (1)$$

meaning that P accepts on all states the same language from u and v if and only if s and t are bisimilar on H. We take

$$\begin{array}{rcl} \text{one new non-terminal } \& & : & \overline{N} \;:=\; N \cup \{\&\} \\ \text{two new terminals } a, b & : & \overline{T} \;:=\; T \cup \{a, b\} \\ \text{one new state } \# & : & \overline{Q} \;:=\; Q \cup \{\#\} \end{array}$$

and we construct the following dpda $\overline{P}$:

$$\overline{P} \;:=\; P \;\cup\; \{\; \#\& \xrightarrow{a} u \;,\; \#\& \xrightarrow{b} v \;\}.$$

By Theorem 4.7 (a), we construct a deterministic graph grammar R generating from a hyperarc $Y \in Dom(R)$, the accessible subgraph $\overline{P}.\overline{N}^*/\#\&$ of $\overline{P}$ from $\#\&$.
Given a graph G, we denote by $\underset{G}{\overset{\epsilon}{\longleftrightarrow}}{}^* \;:=\; (\;\xrightarrow{\epsilon} \cup \xrightarrow{\epsilon}{}^{-1})^*$ the smallest equivalence on its vertices which contains the relation $\xrightarrow{\epsilon} \;:=\; \{\; (s,t) \mid s \xrightarrow{\epsilon} t \;\in G \;\}$ of ϵ-transitions.
We quotient R by identifying vertices linked by an ϵ-transition :

$$\overline{R} \;:=\; \{\; ([X]_{\underset{H}{\overset{\epsilon}{\longleftrightarrow}}{}^*}, [H]_{\underset{H}{\overset{\epsilon}{\longleftrightarrow}}{}^*}) \mid (X, H) \in R \;\} \;.$$

Let H be a graph generated by $\overline{R}$ from $[Y]_{\underset{K}{\overset{\epsilon}{\longleftrightarrow}}{}^*}$ where $(Y, K) \in R$.

Thus H is isomorphic to $[G]_{\underset{G}{\overset{\epsilon}{\longleftrightarrow}}{}^*}$ where $G \;:=\; \overline{P}.\overline{N}^*/\#\&$.

So H is deterministic and of finite out-degree.
Furthermore there are unique vertices r, s, t such that $r \xrightarrow{a} s$ and $r \xrightarrow{b} t$.
Finally H, s, t satisfy property (1).

Another proof is by using Theorem 4.7 (a) and Theorem 5.4 to produce a perfect, standard and entire cf-grammar P' which is deterministic and for every ϵ-rule $fx_1 \ldots x_n \xrightarrow{\epsilon} t$ then

$fx_1...x_n \xrightarrow{a}$ is not possible for every terminal a.
After removing these ϵ-rule $fx_1...x_n \xrightarrow{\epsilon} t$ and by substituing its left hand sides f by its right hand sides t in the remaining right hand sides of P', we obtain a standard and entire cf-grammar without ϵ-rule. Then we apply Theorem 5.3.

b) $\Longrightarrow$ a) : Let R be a deterministic graph grammar and let s_0, t_0 be vertices of a hyperarc $X_0 \in Dom(R)$ such that $R^\omega(X_0)$ is deterministic and of finite out-degree.
We will construct a dpda $\overline{P}$ and two configurations $\overline{u}, \overline{v} \in \overline{Q}.\overline{N}^*$ such that

$$s_0 \equiv_{R^\omega(X_0)} t_0 \quad \text{iff} \quad L(\overline{P}.\overline{N}^*, \overline{u}, \overline{Q}.\overline{N}^*) \ = \ L(\overline{P}.\overline{N}^*, \overline{v}, \overline{Q}.\overline{N}^*) \ .$$

By Theorem 5.3, we can construct a deterministic, standard and entire cf-grammar P on a set F of functions, and two constants $i, j \in F_0$ such that

$$\begin{array}{llll} & P.T(F)^*/i & \text{is isomorphic to} \quad R^\omega(X_0)/s_0 & \text{with } i \text{ corresponding to } s_0 \\ \text{and} & P.T(F)^*/j & \text{is isomorphic to} \quad R^\omega(X_0)/t_0 & \text{with } j \text{ corresponding to } t_0. \end{array}$$

$$\begin{array}{lll} \text{Thus} \ s_0 \equiv_{R^\omega(X_0)} t_0 & \text{iff} & i \equiv_{P.T(F)^*} j \\ & \text{iff} & Label(Path(P.T(F)^*, i)) \ = \ Label(Path(P.T(F)^*, j)), \end{array}$$

by using Lemma A.6. Let m be the maximal arity of the functions defined by P :

$$m \ := \ max\{ \ n \mid \exists \ f, \ fx_1...x_n \in Dom(P) \ \}.$$

Let $\qquad F' \ := \ F \ \cup \ \{ \ \pi_{n,i} \mid 1 \leq n \leq m \ \wedge \ 1 \leq i \leq n \ \}$

where each $\pi_{n,i}$ is of arity n.
Each rule of P is completed to obtain the following perfect, standard and entire cf-grammar P' on F' :

$$\begin{array}{lll} P' & := & \{ \ \pi_{n,i}x_1...x_n \xrightarrow{\epsilon} x_i \mid 1 \leq n \leq m \ \wedge \ 1 \leq i \leq n \ \} \\ & & \cup \ \{ \ fx_1...x_n \xrightarrow{a} [t]_{n,h(t)} \mid fx_1...x_n \xrightarrow{a} t \ \in P \ \} \end{array}$$

where for every $1 \leq i \leq n \leq m$ and every $p \geq 0$,

$$[x_i]_{n,p} \ := \ \begin{cases} x_i & \text{if } \ p = 0 \\ \pi_{n,i}[x_1]_{n,p-1} \ldots [x_n]_{n,p-1} & \text{if } \ p > 0 \ . \end{cases}$$

and for every $f \in F_0$,

$$[f]_{n,p} \ := \ \begin{cases} f & \text{if } \ p = 0 \\ \pi_{1,1}[f]_{n,p-1} & \text{if } \ p > 0 \ . \end{cases}$$

and for every $f \in F - F_0$ with $p > 0$,

$$[ft_1...t_q]_{n,p} \ := \ f[t_1]_{n,p-1} \ldots [t_q]_{n,p-1} \ .$$

So $Label(Path(P'.T(F')^*, t)) \ = \ Label(Path(P.T(F)^*, t))$ for every term t on $F \cup V$.
It suffices to apply Theorem 5.4 to obtain a suitable dpda $\overline{P}$ with suitable configurations $\overline{u}, \overline{v}$.

Let us give another proof by reversing the transformation in $(a) \Longrightarrow (b)$: we add ϵ-transitions to the graph grammar R.
After a possible renaming of vertices, we can assume that the rules of R have distinct vertices: $(V_X \cup V_H) \ \cap \ (V_Y \cup V_K) \ = \ \emptyset$ for every distinct rules $(X, H), (Y, K)$ of R.
We denote by $V \ := \ \bigcup \ \{ \ V_X \mid X \in Dom(R) \ \}$ the vertex set of the left hand sides of R.
We compute the subset $\overline{V} \subseteq V$ of vertices x which are source of an arc in $R^\omega(X)$ for $x \in V_X$ and $X \in Dom(R)$. More exactly $\overline{V}$ is the least fixpoint of the following equation :

$$\overline{V} \ = \ \{ \ p \in V_X \mid \exists \ H, \ (X, H) \in R \ \wedge \ \exists \ a \in T \ \exists \ q, \ p \xrightarrow{a} q \in H \ \}$$

$$\cup \{ \; Y(i) \in V_X \mid \exists\, H,\; (X,H) \in R \;\wedge\; Y \in H \;\wedge\; \exists\, Z \in Dom(R),$$
$$Y(1) = Z(1) \;\wedge\; Z(i) \in \overline{V} \; \} \; .$$

To each vertex $x \in V - \overline{V}$, we associate a new symbol $\sigma(x)$, and we extend σ by the identity for the remaining vertices of R. By applying σ to the right hand sides of R and by adding the rules $\sigma(x) \overset{\epsilon}{\longrightarrow} x$ for $x \in V - \overline{V}$, we obtain the following deterministic graph grammar $\overline{R}$:

$$\overline{R} \;:=\; \{ \; (X \;,\; \sigma(H) \cup \{ \; \sigma(x) \overset{\epsilon}{\longrightarrow} x \mid x \in (V - \overline{V}) \cap V_H \; \}) \mid (X,H) \in R \; \}.$$

Thus for every $X \in Dom(R)$ and $x \in V_X$, we have

$$Label(Path(R^{\omega}(X),x) \;=\; Label(Path(\overline{R}^{\omega}(X),x).$$

So $s_0 \equiv_{R^{\omega}(X_0)} t_0$ iff $Label(Path(\overline{R}^{\omega}(X_0),s_0) \;=\; Label(Path(\overline{R}^{\omega}(X_0),t_0)$.

Note that $\overline{R}^{\omega}(X_0)$ is of finite degree. By Theorem 4.7 (a), we can construct a pushdown automaton P_1 and a configuration $\overline{u}$ such that $P_1.N_1^*/\overline{u}$ is isomorphic to $\overline{R}^{\omega}(X_0)/s_0$ with $\overline{u}$ corresponding to s_0. So P_1 is a dpda.

Similarly we can construct a dpda P_2 and a configuration $\overline{v}$ such that $P_2.N_2^*/\overline{v}$ is isomorphic to $\overline{R}^{\omega}(X_0)/t_0$ with $\overline{v}$ corresponding to t_0.

So $s_0 \equiv_{R^{\omega}(X_0)} t_0$ iff $L(P_1.N_1^*,\overline{u},Q_1.N_1^*) \;=\; L(P_2.N_2^*,\overline{u},Q_2.N_2^*)$.

After a possible renaming, we assume that $N_1 \cap N_2 = \emptyset$.

Finally $\overline{P} := P_1 \cup P_2$ is a suitable dpda.

□

Finally, the deterministic context-free trees corresponds to the algebraic terms, i.e. the finite and infinite terms obtained by prefix unfolding of recursive program schemes.

Lemma 5.10 *The equality problem of deterministic context-free trees is inter-reducible to the equality problem of algebraic terms.*

Proof.

$\Longrightarrow$: Let P be a deterministic term cf-grammar on $F \cup V$, and let s,t be terms on $F \cup V$. We will construct a scheme $\overline{P}$ on $\overline{F} \cup V$, and two terms $\overline{s},\overline{t}$ on $\overline{F}$ such that $Tree(P.T(F)^*,s)$ is isomorphic to $Tree(P.T(F)^*,t)$ if and only if $\overline{P}^{\omega}(\overline{s})$ is isomorphic to $\overline{P}^{\omega}(\overline{t})$.

Let N_P be the set of non-terminals of P :

$$N_P \;:=\; \{ \; f \in F_n \mid fx_1 \ldots x_n \in Dom(P) \; \}.$$

We assume that there is a constant $c \in F_0 - N_P$, otherwise we add a new constant c.

Let $\overline{s} := s[c,\ldots,c]$ and $\overline{t} := t[c,\ldots,c]$.

We choose a strict order $<$ on the set T of terminals (labels of P).

To each function $f \in F_n$, we consider its following label set :

$$T(f) \;:=\; \{ \; a \mid \exists\, t,\; fx_1 \ldots x_n \overset{a}{\longrightarrow} t \; \in P \; \} \; .$$

In particular $T(f) = \emptyset$ for every $f \in F - N_P$. To each subset $T(f)$, we associate a new function $\overline{T(f)}$ of arity $|T(f)|$. We take the following function set $\overline{F}$:

$$\overline{F} \;:=\; F \cup \{ \; \overline{T(f)} \mid f \in F \; \} \; .$$

And we define the following scheme :

$$\overline{P} \;:=\; \{ \; fx_1 \ldots x_n \longrightarrow \overline{T(f)}t_1 \ldots t_m \mid \exists\, a_1,\ldots,a_m,\; fx_1 \ldots x_n \overset{a_i}{\longrightarrow} t_i$$
$$\wedge\; \{a_1,\ldots,a_m\} = T(f) \;\wedge\; a_1 < \ldots < a_m \; \}.$$

In particular for every $f \in F_n - N_P$, $fx_1 \ldots x_n \longrightarrow \overline{\emptyset}$ is a rule of $\overline{P}$.
Given a deterministic tree S and a node u, we associate a (finite or not) term $Term(S,u)$ on $\overline{F}$ as follows :

$$Term(S,u) \;:=\; \overline{\{a_1,\ldots,a_m\}}Term(S,v_1)\ldots Term(S,v_m)$$

where $u \overset{a_i}{\longrightarrow} v_i$ are the arcs of S of source u, and $a_1 < \ldots < a_m$.
Then $Term(Tree(P.T(F)^*,r)) \;=\; \overline{P}^\omega(r[c,\ldots,c])$ for every term $r \in T(F \cup V)$.

Thus $Tree(P.T(F)^*,s)$ is isomorphic to $Tree(P.T(F)^*,t)$
iff $Term(Tree(P.T(F)^*,s)) \;=\; Term(Tree(P.T(F)^*,t))$
iff $\overline{P}^\omega(\overline{s}) \;=\; \overline{P}^\omega(\overline{t})$.

$\Longleftarrow$: Let P be a scheme on $F \cup V$, and let s,t be terms on F.
We will construct a term cf-grammar $\overline{P}$ on $F \cup V$ such that $P^\omega(s)$ is isomorphic to $P^\omega(t)$ if and only if $Tree(\overline{P}.T(F)^*,s)$ is isomorphic to $Tree(\overline{P}.T(F)^*,t)$.
We consider the following set T of terminals (labels of $\overline{P}$ to be constructed) :

$$T \;:=\; (F_0 - N_P) \;\cup\; \{\; g_i \mid \exists\, m,\; g \in F_m - N_P \;\wedge\; 1 \le i \le m \;\}.$$

We take a new constant $-$ (not in F_0), and we construct the following term cf-grammar :

$$\begin{aligned}\overline{P} \;:=\; &\{\; fx_1\ldots x_n \overset{g_i}{\longrightarrow} t_i \mid fx_1\ldots x_n \longrightarrow gt_1\ldots t_m \;\in P \;\wedge\; 1 \le i \le m \;\}\\ &\cup \{\; fx_1\ldots x_n \overset{g}{\longrightarrow} \;-\; \mid fx_1\ldots x_n \longrightarrow g \;\in P \;\}\\ &\cup \{\; gx_1\ldots x_m \overset{g_i}{\longrightarrow} x_i \mid g \in F_m - N_P \;\wedge\; 1 \le i \le m \;\}\\ &\cup \{\; g \overset{g}{\longrightarrow} \;-\; \mid g \in F_0 - N_P \;\}.\end{aligned}$$

By taking $\overline{\{g_1,\ldots,g_m\}} = g$ for every $g \in F_m$ with $m \ge 1$, we define the $Term$ operator as above but with

$$Term(S,u) \;:=\; a \;\; \text{where } u \text{ is source of a unique arc, and its label } a \in F_0 \;.$$

Then $Term(Tree(\overline{P}.T(F)^*,r)) \;=\; P^\omega(r)$ for every term $r \in T(F)$.

Thus $P^\omega(s) \;=\; P^\omega(t)$
iff $Term(Tree(\overline{P}.T(F)^*,s)) \;=\; Term(Tree(\overline{P}.T(F)^*,t))$
iff $Tree(\overline{P}.T(F)^*,s)$ is isomorphic to $Tree(\overline{P}.T(F)^*,t)$.
□

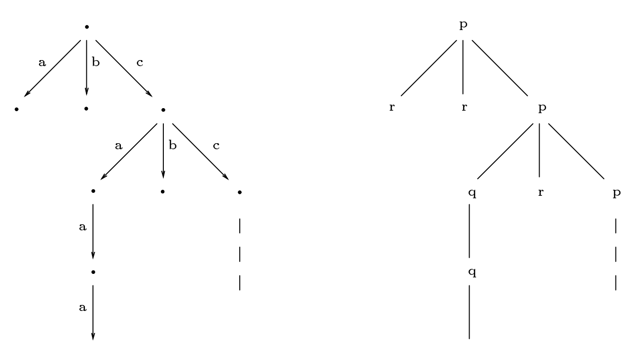
Let us apply the construction of Lemma 5.10 to the following term cf-grammar :

$$P \;=\; \{fx \overset{a}{\longrightarrow} x \;,\; fx \overset{b}{\longrightarrow} gx \;,\; fx \overset{c}{\longrightarrow} fhx \;,\; hx \overset{a}{\longrightarrow} i \;,\; i \overset{a}{\longrightarrow} j\}$$

where $F = \{f,g,h,i,j\}$.
We take the term $t = fx$, and we choose $a < b < c$ and $\overline{\{a,b,c\}} = p$, $\overline{\{a\}} = q$, $\overline{\emptyset} = r$.
From P,t, we obtain the term $\overline{t} = fj$ and the following scheme :

$$\overline{P} \;=\; \{fx \longrightarrow pxgxfhx \;,\; hx \longrightarrow qi \;,\; i \longrightarrow qj \;,\; gx \longrightarrow r \;,\; j \longrightarrow r\} \;.$$

$Tree(P.T(F)^*,t)$ and $Term(Tree(P.T(F)^*,t)) = \overline{P}^\omega(\overline{t})$ are represented as follows :



Let us apply the construction of Lemma 5.10 to the following scheme :

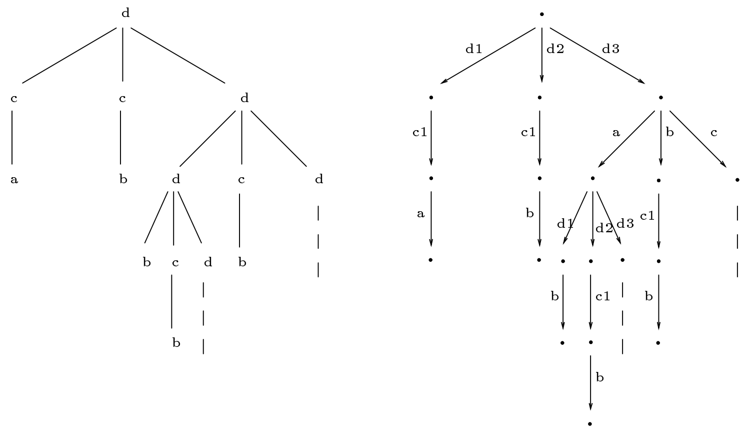
$$P \; = \; \{fx \longrightarrow dxgffb \;,\; g \longrightarrow ch \;,\; h \longrightarrow b\}$$

where $F = \{a, b, c, d, f, g, h\}$ of respective arities $0, 0, 1, 3, 1, 0, 0$.

We take the term $t = fca$. From P, we obtain the following term cf-grammar :

$$\overline{P} \; = \; \{fx \xrightarrow{d_1} x \;,\; fx \xrightarrow{d_2} g \;,\; fx \xrightarrow{d_3} ffb \;,\; g \xrightarrow{c_1} h \;,\; h \xrightarrow{b} - \}$$
$$\cup \; \{dxyz \xrightarrow{d_1} x \;,\; dxyz \xrightarrow{d_2} y \;,\; dxyz \xrightarrow{d_3} z \;,\; cx \xrightarrow{c_1} x \;,\; b \xrightarrow{b} - \;,\; a \xrightarrow{a} - \}.$$

$P^\omega(t) = Term(Tree(\overline{P}.T(F)^*, t))$ and $Tree(\overline{P}.T(F)^*, t)$ are represented as follows :



B Appendix

We study here the entire labelled rewriting systems. And we show that their accessible prefix transition graphs are regular graphs (cf Theorem B.10).

B.a Normalization

Let us begin by normalizing every entire system with an axiom without modifying its accessible prefix rewriting (up to isomorphism). First, we introduce some definitions and notations. Recall that a term t is *ground* if it has no constant, i.e. $F_0(t) = \emptyset$.

Definition B.1 A term t is *normed* if every ground and proper subterm of t is of null height.

In particular, every term of height ≤ 1 is normed. Likewise, every normed term of height > 1 is not ground. By extension, a *normed system* is a system whose rule terms are normed. We say that a system R is a *proper system* if the left hand sides of its rules are not variables, *i.e.* for every rule $s \longrightarrow t$ of R, $s \notin V$. Let

$$h(R) \quad = \quad max\{ \ h(s) \mid \exists t, \ (s \longrightarrow t) \in R \quad \vee \quad (t \longrightarrow s) \in R \ \}$$

the *height of a system* R, *i.e.* the maximal height of its terms. Likewise, we denote by $h(\sigma) = \{ \ h(\sigma(x)) \mid x \in V \ \}$ the *height of a substitution* σ. Let us recall that a *constant* is a symbol of F_0. Let $E(R)$ be the set of symbols in $E \subseteq F \cup V$ of the terms of a system R, *i.e.*

$$E(R) \quad = \quad \bigcup \{ \ E(s) \mid \exists t, \ (s \longrightarrow t) \in R \quad \vee \quad (t \longrightarrow s) \in R \ \}.$$

We denote by $\underset{R}{\overset{a}{\longmapsto}}$ the relation of prefix transitions labelled by a, according to any system R : $s \underset{R}{\overset{a}{\longmapsto}} t$ if $s \overset{a}{\longrightarrow} t \ \in R.T(F)^*$, i.e.

$s \overset{a}{\longrightarrow} t$ if $s = \sigma(s_0)$ and $t = \sigma(t_0)$ for some $s_0 \overset{a}{\longrightarrow} t_0 \ \in R$ and some substitution σ.

Furthermore $\underset{R}{\longmapsto} \ := \ \bigcup_a \ \underset{R}{\overset{a}{\longmapsto}}$ is the *prefix rewriting* according to R, and its reflexive and transitive closure $\underset{R}{\longmapsto}^*$ is the *prefix derivation* of R.

Let $|A|$ be the cardinality of a set A. Given a binary relation R, we denote by $Dom(R) = \{ \ s \mid \exists \ t, \ s \ R \ t \ \}$ the *domain* of R and by $Im(R) = \{ \ t \mid \exists \ s, \ s \ R \ t \ \}$ the *image* of R. Let $f(R) = \{ \ (f(s), f(t)) \mid s \ R \ t \ \}$ the *transformation* of a binary relation R on a set A by a mapping f from A into A. It is now possible to give a normalization of the entire systems.

Proposition B.2 *Any pair* (R, r) *consisting of a labelled entire system* R *and a term* r *may be effectively transformed into another pair* (S, s) *of a proper, normed and entire labelled system* S *of height* $h(S) \leq 2$ *and a constant* $s \in F_0$ *such that* $S.T(F)^*/s$ *is effectively isomorphic to* $R.T(F)^*/r$.

Proof.
The method given here is a generalization of Lemma 2.4 of [MS 85].

To avoid identical rules up to variables renaming, we order the set $V = \{x_1, x_2, \ldots\}$ of variables. We may suppose R proper and $V(R) = \emptyset$. Otherwise, we take an injection σ from $V(r)$ into $F_0 - F(R)$ and we replace (R, r) by $(S, \sigma(r))$ where

$$S \;=\; \{\; (s \xrightarrow{a} t) \in R \mid s \notin V \;\} \;\;\cup\;\; \{\; (fx_1\ldots x_p \xrightarrow{a} t[x \mapsto fx_1\ldots x_p]) \mid$$
$$(x \xrightarrow{a} t) \in R \;\;\wedge\;\; x \in V \;\;\wedge\;\; f \in (F(R) \cup Im(\sigma)) \cap F_p \;\}.$$

So S is proper, $V(\sigma(r)) = \emptyset$ and $R'.T(F)^*/\sigma(r) \;=\; \sigma(R.T(F)^*/r)$.
We denote by

$$m \;=\; max(\;\{\; h(s) \mid \exists\, a, t,\; (s \xrightarrow{a} t) \in R \;\;\vee\;\; (t \xrightarrow{a} s) \in R \;\} \cup \{h(r)\}\;)$$

the maximal height of r and of the terms of R. We may suppose $m \neq 0$ otherwise (R, r) is suitable. We consider the smallest set T of (linear) terms such that

$$\begin{array}{lrcl} & F_0 \cup \{x_1\} & \subseteq & T \\ \text{and} & ft_1\sigma_1(t_2)\ldots\sigma_{p-1}(t_p) & \in & T \;\text{ for every } f \in F_p \text{ and } t_1, \ldots, t_p \in T, \end{array}$$

where for every $1 \leq i \leq p$, σ_i is the substitution defined by

$$\sigma_i(x_j) \;=\; x_k \quad \text{for} \quad k = j + |V(t_1)| + \ldots + |V(t_i)| \;.$$

To every substitution σ, we associate the set $D_\sigma = \{\; x \in V \mid \sigma(x) \neq x \;\}$ of non fixed variables of σ, and the sequence $h(\vec{\sigma}) = h(\sigma(x_1))h(\sigma(x_2))\ldots \in \mathbb{N}^\infty$ of the heights of the terms substituted to the variables.
Take an injection I from $\{\; t \in T - \{x_1\} \mid h(t) \leq m \;\}$ into F such that $I(t) \in F_{|V(t)|}$ and $I(fx_1\ldots x_p) = f$. The image of I is denoted by $[F]$. We define a mapping [ ] from $T(F(R) \cup V)$ into $T([F] \cup V)$ by

$$\begin{array}{lrcll} & [t] & = & t & \text{if } h(t) = 0 \\ \text{and} & [t] & = & I(s)[\sigma(x_1)]\ldots[\sigma(x_{|V(s)|})] & \text{if } h(t) \neq 0 \text{ where} \\ & & & & s \in T - \{x_1\} \,, h(s) \leq m \,, \sigma(s) = t \text{ and } h(\vec{\sigma}) \in (m\mathbb{N})^\infty. \end{array}$$

The mapping [ ] is well defined: given a term t, there exist a unique term s of $T - \{x_1\}$ and a unique substitution σ with $D_\sigma \subseteq \{x_1, \ldots, x_{|V(s)|}\}$ such that $h(s) \leq m$, $\sigma(s) = t$ and $h_\sigma \in (m\mathbb{N})^\infty$.
Furthermore [ ] is injective, and for every term t, $V([t]) = V(t)$.
We denote by $x(u) = min\{\; \{\infty\} \cup \{\; i \mid u(i) = x \;\}\;\}$ the smallest occurrence of a letter x in a word u. Let

$$P \;=\; \{\; t \in T(F \cup V) \mid x_i(t) \leq x_{i+1}(t) \text{ for every } i \geq 1 \;\}$$

be the set of terms where no variable appears (by prefix) before the preceding variables have already appeared.
We define the following system S:

$$S \;=\; \{\; [\sigma(s)] \xrightarrow{a} [\sigma(t)] \mid (s \xrightarrow{a} t) \in R \;\wedge\; \sigma \in Subst \;\wedge\; \sigma(s) \in P \;\wedge\; h(\sigma) < m \;\}.$$

By definition of P, S has only a finite number of rules. Moreover, every constant of a term t of S is at height 0 or 1 of t, so S is normed. Furthermore $h(S) \leq 2$. First, we show (point i) that S is entire. Then we prove (point iii) and (point iv) that $S.T(F)^*/[r] \;=\; [R.T(F)^*/r]$, hence $(S,[r])$ is suitable.

i) Let us show that S is entire. It suffices to establish the following property:

$$\text{if } \; t \text{ is entire and } h(\sigma) < m \; \text{ then } \; [\sigma(t)] \text{ is entire.}$$

By induction on $h(t) \geq 0$.

If $h(t) = 0$ then $h(\sigma(t)) < m$ so $h[\sigma(t)] \leq 1$ hence $[\sigma(t)]$ is entire.
If $h(t) > 0$ so $h(\sigma(t)) \neq 0$ hence

$$[\sigma(t)] = I(s)[\tau(x_1)]\ldots[\tau(x_{|V(s)|})] \;\text{ with}$$
$$s \in T - \{x_1\} \;,\; h(s) \leq m \;,\; \tau(s) = \sigma(t) \text{ and } h(\vec{\tau}) \in (m\mathbb{N})^\infty.$$

Let $1 \leq i \leq |V(s)|$ such that $h[\tau(x_i)] \neq 0$. So $h(\tau(x_i)) \geq m$. As $h(\sigma) < m$, there exists an occurrence u of t such that $t(u) \notin V$ and $\tau(x_i) = \sigma(t\backslash u)$.
If $V(t\backslash u) = \emptyset$ then $V[\tau(x_i)] = V[\sigma(t\backslash u)] = \emptyset$.
If $V(t\backslash u) \neq \emptyset$ then $V(t\backslash u) = V(t)$ because t is entire. So $V(\sigma(t\backslash u)) = V(\sigma(t))$ hence

$$V[\tau(x_i)] = V(\tau(x_i)) = V(\sigma(t\backslash u)) = V(\sigma(t)) = V[\sigma(t)].$$

Furthermore $t\backslash u$ is entire, hence by induction hypothesis $[\tau(x_i)] = [\sigma(t\backslash u)]$ is entire. Finally $[\sigma(t)]$ is entire and this terminates the induction.

ii) Let us show that every term t and every substitution σ satisfy the following property:

$$[\sigma(t)] \;=\; [\sigma]([t]) \quad if \;\, h(\vec{\sigma}) \in (m\mathbb{N})^\infty \tag{1}$$

where $[\sigma]$ is the substitution defined by $[\sigma](x) = [\sigma(x)]$ for every $x \in V$.
Let us prove (1) by induction on $h(t) \geq 0$.
$h(t) = 0$: $t \in F_0 \cup V$ so $[\sigma(t)] = [\sigma](t) = [\sigma]([t])$.
Let us show (1) for $h(t) > 0$ supposing that (1) is true for every term of height $< h(t)$.
Let us take the term s of $T - \{x_1\}$ and the substitution τ such that $\tau(s) = t$, $h(s) \leq m$, $D_\tau \subseteq V(s)$ and $h(\vec{\tau}) \in (m\mathbb{N})^\infty$. We define the composition $\sigma{\circ}\tau$ by $(\sigma{\circ}\tau)(x) = \sigma(\tau(x))$ for every $x \in V$. So

$$\begin{aligned}
[\sigma(t)] &= [\sigma(\tau(s))] = [(\sigma{\circ}\tau)(s)] \\
&= I(s)[(\sigma{\circ}\tau)(x_1)]\ldots[(\sigma{\circ}\tau)(x_{|V(s)|})] && \text{because } h(\vec{\sigma{\circ}\tau}) \in (m\mathbb{N})^\infty \\
&= I(s)[\sigma(\tau(x_1))]\ldots[\sigma(\tau(x_{|V(s)|}))] \\
&= I(s)[\sigma]([\tau(x_1)])\ldots[\sigma]([\tau(x_{|V(s)|})]) && \text{by ind. hyp. on } h(\tau(x_i)) < h(t) \\
&= [\sigma](I(s)[\tau(x_1)]\ldots[\tau(x_{|V(s)|})]) && \text{because } [\sigma] \text{ is a substitution} \\
&= [\sigma]([\tau(s)]) && \text{by definition of } [\;] \\
&= [\sigma]([t]).
\end{aligned}$$

This ends the induction, so property (1) is proved.

iii) Let us show that $[R.T(F)^*/r] \subseteq S.T(F)^*/[r]$. It suffices to prove that

$$s \underset{R}{\overset{a}{\longmapsto}} t \implies [s] \underset{S}{\overset{a}{\longmapsto}} [t] \quad \text{for every term } s \text{ and } t \text{ on } F \cup V.$$

Let s and t be terms such that $s \underset{R}{\overset{a}{\longmapsto}} t$. So, there exist a rule $s_0 \overset{a}{\longrightarrow} t_0$ of R and a substitution σ_0 such that $s = \sigma_0(s_0)$ and $t = \sigma_0(t_0)$.
We decompose σ_0 into $\sigma_0 = \tau{\circ}\sigma$ such that $h(\sigma) < m$ and $h(\vec{\tau}) \in (m\mathbb{N})^\infty$. Up to a variables renaming, we suppose that $\sigma(s_0) \in P$. So $[\sigma(s_0)] \overset{a}{\longrightarrow} [\sigma(t_0)]$ is a rule of S, hence from (ii)

$$[s] = [\tau(\sigma(s_0))] = [\tau]([\sigma(s_0)]) \underset{S}{\overset{a}{\longmapsto}} [\tau]([\sigma(t_0)]) = [\tau(\sigma(t_0))] = [t];$$

this ends the proof of (iii).

iv) Let us show that $S.T(F)^*/[r] \subseteq [R.T(F)^*/r]$. It suffices to show that

$$[s] \underset{S}{\overset{a}{\longmapsto}} t' \implies \exists t,\ s \underset{R}{\overset{a}{\longmapsto}} t \ \wedge\ [t] = t' \quad \text{for every } s \in T(F{\cup}V) \text{ and } t' \in T([F]{\cup}V).$$

Let $s \in T(F{\cup}V)$ and consider a transition $[s] \underset{S}{\overset{a}{\longmapsto}} t'$. So, there exist a rule $s_0 \overset{a}{\longrightarrow} t_0$ of R, a substitution σ satisfying $\sigma(s_0) \in P$ and $h(\sigma) < m$, and a substitution τ such that $[s] = \tau([\sigma(s_0)])$ and $t' = \tau[\sigma(t_0)]$.
We can suppose that the respective sets D_τ and D_σ of non fixe variables of τ and σ are restricted as follows:

$$D_\tau = V[\sigma(s_0)] \cup V[\sigma(t_0)] = V[\sigma(s_0)] = V(\sigma(s_0)) = D_\sigma.$$

Let us show that there exists a substitution μ such that $h(\vec{\mu}) \in (m\mathbb{N})^\infty$ and $[\mu] = \tau$.

Either	$h(s_0) = 0$.	As R is proper, $s_0 \in F_0$ so $[\sigma(s_0)] = [s_0] = s_0$. hence $D_\tau = \emptyset$ and the identity for μ suits, *i.e.* $D_\mu = \emptyset$.
Or	$h(s_0) > 0$.	Then $h(\sigma(s_0)) \geq h(s_0) > 0$. So, the existence of μ is deducible from $[s] = \tau([\sigma(s_0)])$.

From (ii) $[s] = [\mu]([\sigma(s_0)]) = [\mu(\sigma(s_0))]$ and as $[\ ]$ is injective, we have $s = \mu(\sigma(s_0))$. Similarly $t' = [\mu]([\sigma(t_0)]) = [\tau(\sigma(t_0))]$. Hence $t = \tau(\sigma(t_0))$ suits and (iv) is proved.
□

Contrary to Theorem B.10, let us remark that this normalization of an entire system does not give an entire cf-grammar.

B.b Prefix derivation

First, let us give three basic properties concerning ground and normed terms.
Every subterm of an entire term is entire. More generally, the class of entire terms is closed by prefix derivation according to any entire system.

Lemma B.3 *Given an entire system R, if $s \underset{R}{\longmapsto}^* t$ and s is entire then t is entire.*

Proof.
By induction on the length of the prefix derivation, it suffices to establish it for $s \underset{R}{\longmapsto} t$ with s entire. There exist a rule $s_0 \longrightarrow t_0$ of R and a substitution σ such that $s = \sigma(s_0)$ and $t = \sigma(t_0)$. Let t' be a subterm of t such that $t' \notin V$ and $V(t') \neq \emptyset$. To show that t is entire, it suffices to show that $V(t') = V(t)$. We distinguish the two cases below.
Case 1: there is a variable x of t_0 such that t' is a subterm of $\sigma(x)$.

So t' and $\sigma(x)$ are subterms of s, with at least one variable, but not reduced to this variable. As s is entire, we have $V(t') = V(\sigma(x)) = V(s)$. So
$V(t) \subseteq V(s) = V(t') \subseteq V(t)$ hence $V(t') = V(t)$.

Case 2: there is a subterm $t'_0 \notin V$ of t_0 such that $t' = \sigma(t'_0)$.

As $V(t') \neq \emptyset$, we have $V(t'_0) \neq \emptyset$. So $V(t'_0) = V(t_0)$ because R is entire and therefore t_0 is entire.

In consequence $V(t') = V(\sigma(t'_0)) = V(\sigma(t_0)) = V(t)$.

In all cases $V(t') = V(t)$. Finally t is entire.

□

Similarly, the prefix derivation according to any entire and normed system preserves the normality of non ground terms.

Lemma B.4 *Given an entire and normed system R, if $s \underset{R}{\longmapsto}^* t$ where s is normed and t is not ground, then t is normed.*

Proof.

By induction on the length of the prefix derivation, it suffices to show it for $s \underset{R}{\longmapsto} t$ with s normed and $V(t) \neq \emptyset$. There exist a rule $s_0 \longrightarrow t_0$ of R and a substitution σ such that $s = \sigma(s_0)$ and $t = \sigma(t_0)$. Let t' be a proper and ground subterm of t. To show that t is normed, it suffices to prove that $h(t') = 0$. We distinguish the two cases below:

Case 1: there is a variable x of t_0 such that t' is a subterm of $\sigma(x)$.

Thus t' is a subterm of s. As $V(t) \neq \emptyset$, we have $V(s) \neq \emptyset$ hence $t' \neq s$. Thus $h(t') = 0$ because s is normed.

Case 2: there is a subterm $t'_0 \notin V$ of t_0 such that $t' = \sigma(t'_0)$.

Case 2.1: $V(t'_0) = \emptyset$.

Thus $t' = t'_0$. As $t' \neq t$, we have $t' \neq t_0$. So $h(t') = 0$ because t_0 is normed.

Case 2.2: $V(t'_0) \neq \emptyset$.

As t_0 is entire, we have $V(t'_0) = V(t_0)$. Consequently $\emptyset = V(t') = V(\sigma(t'_0)) = V(\sigma(t_0)) = V(t)$; which is excluded by hypothesis.

In any case $h(t') = 0$. Finally, t is normed.

□

Let σ_t be the *restriction of a substitution* σ to the variables of a term t, *i.e.* for every variable x of t, $\sigma_t(x) = \sigma(x)$ else σ_t is the identity. This notation allows to express a basic property concerning the substitution of an entire and normed term.

Lemma B.5 *Given a substitution σ and an entire and normed term t, we have*

$$h(\sigma(t)) = h(t) + h(\sigma_t) \ .$$

Proof.

By induction on $h(t) \geq 0$.

$h(t)=0:$ let $t \in F_0$ so $\sigma(t)=t$ and $h(\sigma_t)=0$; hence the equality.
let $t \in V$ so $h(t)=0$ and $h(\sigma(t))=h(\sigma_t)$; hence the equality.
$h(t)=1:$ $h(\sigma(t))=1+h(\sigma_t)=h(t)+h(\sigma_t)$.
$h(t)>1:$ we can decompose t into $t=ft_1...t_p$ where $f \in F$.
Therefore $\sigma(t)=f\sigma(t_1)...\sigma(t_p)$.

$$\text{Thus } h(\sigma(t)) = 1+max\{h(\sigma(t_1)),\ldots,h(\sigma(t_p))\}$$
$$= 1+max\{h(t_1)+h(\sigma_{t_1}),\ldots,h(t_p)+h(\sigma_{t_p})\} \text{ by induction hyp.}$$

Take a q such that $h(t_q)$ is maximal.
So $h(t_q) \geq 1$ and as t is entire and normed, we have $V(t_q)=V(t)$.
Thus $h(\sigma_{t_q})=h(\sigma_t)$ is maximal, hence

$$h(\sigma(t))=1+h(t_q)+h(\sigma_{t_q})=h(t)+h(\sigma_t).$$

This ends the induction and the proof of this lemma.
□

We are now able to decide if a given term derives by prefix from another one according to an entire system.

Proposition B.6 *The prefix derivation of an entire system is decidable.*

i) As the labels do not matter in this proposition, we will not take them into account. Let R be an entire system and let us consider the terms s and t. Let us show that we can decide if $s \underset{R}{\longmapsto}^* t$.
Add the rules $s \longrightarrow s$ and $t \longrightarrow t$ in R. From Proposition B.2, we can suppose R proper, normed, of height $h(R) \leq 2$ and $s,t \in F_0 \cap Dom(R)$. Without modifying the decision of the prefix derivation of s in t, we can suppose that R is completed into the following system:

$$R \cup \{\ \sigma(u) \longrightarrow \sigma(v) \mid (u \longrightarrow v) \in R\ \wedge\ \sigma \in Subst \wedge h(\sigma)=0 \wedge \sigma(u) \in P \wedge V(\sigma(v)) \neq \emptyset\} \tag{1}$$

where the set $V=\{x_1,x_2,\ldots\}$ of variables is ordered and P is the set, defined in the proof of Proposition B.2, of the terms in which no variable x_i occurs (by prefix) without previous occurrence of each of the variables $x_1,\ldots,x_{i-1}$. So completed, R has only a finite number of rules, and remains normed and entire. Let us point out that

$$s \underset{R}{\longmapsto}^* t \quad \text{iff} \quad (s \longrightarrow t) \in S$$

where $$S\ =\ \{\ s \longrightarrow t \mid s\ R \circ \underset{R}{\longmapsto}^* t\ \ \wedge\ \ h(t) \leq 1\ \}.$$
Thus $s \underset{R}{\longmapsto}^* t$ is decidable iff S is constructible.

ii) Let us show that S is constructible. For every system U, we define
$$\underline{U}\ =\ \{\ (s \longrightarrow t) \in U \mid h(t) \leq h(s)\ \}$$
and $$\underset{U}{\Longrightarrow}\ =\ \underset{\underline{U}}{\longmapsto}^* \cup (\underset{\underline{U}}{\longmapsto}^* \circ \underset{U}{\longmapsto})$$

the operation deriving by prefix according to $\underline{U}$ followed or not by the prefix rewriting according to U. Thus $\underset{U}{\Longrightarrow}$ is decidable. To construct S, we define the following sequence $(S_n)_{n\geq 0}$ of systems:

$$\begin{array}{rcl} S_0 & = & \emptyset \\ S_{n+1} & = & \{ \; s \longrightarrow t \mid s \; R \circ \underset{S_n}{\Longrightarrow} t \; \wedge \; h(t) \leq 1 \; \} \end{array}$$

By induction on $n{\geq}0$, $S_n \subseteq S_{n+1} \subseteq R \circ \underset{R}{\longmapsto}^*$; hence $S_n \subseteq S$ for every $n{\geq}0$. So the integer $q = min\{ \; n \mid S_n = S_{n+1} \; \}$ exists. Moreover and by induction on $n{\geq}0$, $S_{q+n} = S_q$. Let us show that $S_q = S$. It remains to prove that $S \subseteq S_q$. For this and by induction on $n{\geq}0$, it suffices to establish the following property:

$$s \underset{R}{\longmapsto}^n t \; \wedge \; h(t) \leq 1 \quad \Longrightarrow \quad s \underset{S_n}{\Longrightarrow} t \; . \tag{2}$$

By induction on $n{\geq}0$.

$n = 0:$

$s = t$ so $s \underset{\emptyset}{\Longrightarrow} t$ because $\underset{\emptyset}{\Longrightarrow} = \underset{\emptyset}{\longmapsto}^0 = \{ \; (s,s) \mid s \in T(F{\cup}V) \; \}$.

$(\leq n) \Longrightarrow n+1:$

Let $s \underset{R}{\longmapsto}^{n+1} t$ such that $h(t) \leq 1$. Consider the following integer

$$m \; = \; max\{ \; i \mid \exists \; s_0, u, \sigma, \; s_0 \; R \circ \underset{R}{\longmapsto}^i u \; \wedge \; \sigma(s_0) = s \; \wedge \; \sigma(u) \underset{R}{\longmapsto}^{n\text{-}i} t \; \}$$

We distinguish the two following complementary cases:

Case 1: $m = n$

By definition of m, there exists s_0, t_0, v, σ such that

$s_0 \; R \; v \underset{R}{\longmapsto}^n t_0 \; , \; \sigma(s_0) = s \; , \; \sigma(t_0) = t \; .$

As $h(t_0) \leq h(t) \leq 1$, we have $v \underset{S_n}{\Longrightarrow} t_0$ by induction hypothesis.

Thus $s_0 \; S_{n+1} \; t_0$ therefore $s \; = \; \sigma(s_0) \underset{S_{n+1}}{\Longrightarrow} \sigma(t_0) \; = \; t$.

Case 2: $m < n$

There exist s_0, u, σ such that $s_0 \; R \circ \underset{R}{\longmapsto}^m u \; , \; \sigma(s_0) = s \; , \; \sigma(u) \underset{R}{\longmapsto}^{n\text{-}m} t$.

As $m < n$, we have $V(u) \neq \emptyset$. From the completion (1), we can suppose that s_0 and σ satisfy the following property:

$$\sigma(x) \notin F_0 \; \text{ and } \; \sigma(x) \neq \sigma(y) \quad \text{for every } \; x, y \in V(s_0) \; \text{ with } x \neq y. \tag{3}$$

Let us show that $h(u) \leq 1$. As $n - m > 0$, there exists v such that

$\sigma(u) \underset{R}{\longmapsto} v \underset{R}{\longmapsto}^{n\text{-}m\text{-}1} t$

Thus there exist a rule $s_1 \longrightarrow t_1$ of R and a substitution τ such that $\tau(s_1) = \sigma(u)$ and $\tau(t_1) = v$. Assume that $h(u) \geq 2$.

As $V(u) \neq \emptyset$ and from Lemma B.3 and Lemma B.4, u is an entire and normed term. From Lemma B.5, we have

$2 + h(\sigma_u) \leq h(u) + h(\sigma_u) = h(\sigma(u)) = h(\tau(s_1)) = h(s_1) + h(\tau_{s_1}) \leq 2 + h(\tau_{s_1})$

hence $h(\sigma_u) \leq h(\tau_{s_1})$. As $h(\tau(s_1)) = h(\sigma(u)) \geq 2$, we have $s_1 \notin F_0$ so $h(s_1) > 0$ because R is proper. We can write $s_1 = f w_1 \ldots w_p$. As $h(u) \geq 2$ and $\sigma(u) = \tau(s_1) = f\tau(w_1)\ldots\tau(w_p)$, we can write $u = f u_1 \ldots u_p$.

Let us show that there exists a substitution μ such that $\mu(s_1) = u$.

From property (3), it suffices to prove that $w_j \in V$ when $u_j \in V$. Assume that there exists a $1 \leq j \leq p$ such that $u_j \in V$. Then
$h(\tau(w_j)) = h(\sigma(u_j)) \leq h(\sigma_u) \leq h(\tau_{s_1})$.
As w_j is a subterm of s_1 entire and normed, we have $h(w_j) = 0$. From property (3), $h(\tau(w_j)) = h(\sigma(u_i)) > 0$ hence $w_j \in V$.

Therefore $u = \mu(s_1) \underset{R}{\longmapsto} \mu(t_1)$. As $\tau(s_1) = \sigma(u) = \sigma(\mu(s_1))$, we have $\sigma \circ \mu_{s_1} = \tau_{s_1}$. So $\sigma(\mu(t_1)) = \tau(t_1) = v$ because $V(t_1) \subseteq V(s_1)$.
Finally $s_0\ R \circ \underset{R}{\longmapsto}^{m+1} \mu(t_1)$ and $\sigma(\mu(t_1)) \underset{R}{\longmapsto}^{n-m-1} t$; which is in contradiction with the maximality of m.
Finally $h(u) \leq 1$. By induction hypothesis, we have
$s_0\ R \circ \underset{S_m}{\Longrightarrow} u$ so $s_0\ S_{m+1}\ u$. Consequently $s_0\ S_n\ u$. As $V(u) \neq \emptyset$, we have $V(s_0) \neq \emptyset$ so $h(s_0) \geq 1 \geq h(u)$ hence $s_0\ \underline{S_n}\ u$. Moreover and by induction hypothesis, $\sigma(u) \underset{S_{n-m}}{\Longrightarrow} t$ so $\sigma(u) \underset{S_n}{\Longrightarrow} t$.
Finally $s = \sigma(s_0) \underset{\underline{S_n}}{\longmapsto} \circ \underset{S_n}{\Longrightarrow} t$.
As $\underset{\underline{S_n}}{\longmapsto} \circ \underset{S_n}{\Longrightarrow} \subseteq \underset{S_n}{\Longrightarrow}$, we have $s \underset{S_n}{\Longrightarrow} t$ hence $s \underset{S_{n+1}}{\Longrightarrow} t$.

This ends the induction and proves property (2). Thus S is constructible, so $s \underset{R}{\longmapsto}^* t$ is decidable.
$\square$

Let us indicate that we can show that the prefix derivation of any context-free grammar is decidable.

B.c Transformation

We need some basic properties on the terms obtained by prefix derivation according to an entire and normed system.
The loss of variables by prefix rewriting brings a fall of height.

Lemma B.7 *Given an entire and normed system R, if $s \underset{R}{\longmapsto} t$ with s entire and normed and $V(s) \neq V(t)$ then $h(t) \leq h(R)$.*

Proof.
There exists a rule $s_0 \longrightarrow t_0$ of R and a substitution σ such that $\sigma(s_0) = s$ and $\sigma(t_0) = t$. As R is entire and normed, and from Lemma B.5, we have

$$h(t) = h(t_0) + h(\sigma_{t_0}) \leq h(R) + h(\sigma_{t_0}) \ .$$

If there is a variable x of t_0 then x is a variable of s_0 and satisfies one of the three conditions below.

Case 1: $\sigma(x) \in V$. Thus $h(\sigma(x)) = 0$

Case 2: $V(\sigma(x)) = \emptyset$.
As $V(s) \neq V(t)$, we have $V(s) \neq \emptyset$. Therefore $s \neq x$. Hence $\sigma(x)$ is a proper and ground subterm of s. As s is normed, $h(\sigma(x)) = 0$.

Case 3: $\sigma(x) \notin V$ and $V(\sigma(x)) \neq \emptyset$.

As s is entire, $V(\sigma(x)) = V(s)$. From Lemma B.3, t is entire. As $\sigma(x)$ is a subterm of t, we have $V(\sigma(x)) = V(t)$. Finally $V(s) = V(t)$ which is forbidden by hypothesis.

In any case, $h(\sigma(x)) = 0$ for every $x \in V(t_0)$, *i.e.* $h(\sigma_{t_0}) = 0$ hence $h(t) \leq h(R)$.
□

Let us give a sufficient condition for a subterm of a substitution of an entire and normed term to be a subterm of one of the substituted terms.

Lemma B.8 *Given a substitution σ and a term s entire and normed, if t is a subterm of $\sigma(s)$ of height $0 < h(t) \leq h(\sigma_s)$ then there is a variable x of s such that t is a subterm of $\sigma(x)$.*

Proof.
By contraposition. Let s be an entire and normed term, and σ a substitution. Let t be a subterm of $\sigma(s)$ of height $h(t) \neq 0$, and not a subterm of an $\sigma(x)$ for $x \in V(s)$. Thus there exists a subterm $s_0 \notin V$ of s such that $t = \sigma(s_0)$. As s is normed and $h(t) \neq 0$, we have $V(s_0) \neq \emptyset$. Thus $V(s_0) = V(s)$ because s is entire. Hence $V(t) = V(\sigma(s_0)) = V(\sigma(s))$. As $h(t) \neq 0$, $h(t) \geq 1 + h(\sigma_s)$.
□

We say that a system R is a *normalized system* if R is entire, normed and proper, and of height $h(R) \leq 2$. Recall that Proposition B.2 transforms every entire system and every term into a normalized system R and a constant r with an isomorphic accessible prefix rewriting graph. For every term s accessible by prefix derivation according to R from r, we will show that the subterms of s of the same height have the same set of immediate subterms, except for the constants. More exactly, we denote by $T(ft_1...t_p) = \{t_1, \ldots, t_p\} - F_0$ the set of immediate subterms of $ft_1...t_p$ (where $f \in F \cup V$) which are not constants. In a more general way and for every subterm t of s, we show that every subterm of s not in F_0 and of maximal height $< h(t)$ is an immediate subterm of t. Formally, we denote by

$$E(s,n) \;=\; \{\, t \notin F_0 \mid \exists\, m,\ t \in T(s\backslash m) \;\wedge\; h(t) < n \leq h(s\backslash m) \,\} \quad \text{for } n \leq h(s)$$

the set of the subterms of s of maximal height less than n which are not constants.

Lemma B.9 *Given a normalized system R and a constant r, if $r \underset{R}{\longmapsto}^{*} s$ then $E(s, h(t)) = T(t)$ for every subterm t of s.*

Proof.
By definitions of T and E, we have $T(t) \subseteq E(s, h(t))$. Furthermore,
(a) $E(s, h(s)) = T(s)$
(b) $E(s, 0) = \emptyset = T(t)$ for every subterm t of s of null height.
Let us show the inverse inclusion $E(s, h(t)) \subseteq T(t)$ by induction on the derivation length from r to s.
As $r \in F_0$ and from (b), r satisfies the inclusion.

Suppose that $r \underset{R}{\longmapsto}^{*} s \underset{R}{\longmapsto} t$ and that s satisfies the inclusion. Let t' be a subterm of t and let us show that $E(t, h(t')) \subseteq T(t')$. From (a) and (b), we can suppose that $0 < h(t') \leq h(t) - 1$. Let $v \in E(t, h(t'))$. In particular $v \notin F_0$, and there exists a subterm u of t such that $v \in T(u)$ and $h(v) < h(t') \leq h(u)$. To establish this lemma, it suffices to show that $v \in T(t')$. As $s \underset{R}{\longmapsto} t$, there exist a rule $s_0 \longrightarrow t_0$ of R and a substitution σ such that $\sigma(s_0) = s$ and $\sigma(t_0) = t$. As R is entire and normed, s_0 and t_0 are entire and normed. As R is proper, normed and of height $h(R) \leq 2$, we have $h(t_0) \leq h(s_0) + 1$. Thus and from Lemma B.5, we have

$$h(t') \;\leq\; h(t) - 1 \;=\; h(t_0) + h(\sigma_{t_0}) - 1 \;\leq\; h(s_0) + h(\sigma_{t_0}) \;\leq\; h(s_0) + h(\sigma_{s_0}) \;=\; h(s).$$

Hence $E(s, h(t'))$ exists. Note that

$$h(v) \;\leq\; h(t') - 1 \;\leq\; h(t) - 2 \;\leq\; h(t) - h(t_0) \;=\; h(\sigma_{t_0}).$$

We distinguish the two cases below.

Case 1: $h(u) \leq h(\sigma_{t_0})$.

From Lemma B.8, there exists a variable x of t_0 such that u is a subterm of $\sigma(x)$. As $V(t_0) \subseteq V(s_0)$, u is a subterm of $\sigma(s_0) = s$. Thus $v \in E(s, h(t'))$.
As $h(t) \leq h(u) \leq h(\sigma_{t_0})$ and similarly, t' is a subterm of s. By induction hypothesis $E(s, h(t')) \subseteq T(t')$ hence $v \in T(t')$.

Case 2: $h(u) > h(\sigma_{t_0})$.

So, there exists a subterm $t'_0 \notin V$ of t_0 such that $u = \sigma(t'_0)$. As $v \in T(u) - F_0$ and t_0 is normed, we have $V(t'_0) \neq \emptyset$. Therefore $V(t'_0) = V(t_0)$ because t_0 is entire. As $h(v) \leq h(\sigma_{t_0})$, there exists a variable x of t_0 such that $v = \sigma(x)$. As R is proper, there exists a subterm w of s_0 such that $x \in T(w)$ and $V(w) = V(s_0)$. Then $v \in T(\sigma(w))$ and

$$h(\sigma(w)) \;\geq\; 1 + h(\sigma_{s_0}) \;\geq\; 1 + h(\sigma_{t_0}) \;\geq\; h(t_0) - 1 + h(\sigma_{t_0}) \;=\; h(t) - 1 \;\geq\; h(t').$$

Hence $v \in E(s, h(t'))$. We consider the two following subcases.

Case 2.1: $h(t') \leq h(\sigma_{t_0})$.

From Lemma B.8, t' is a subterm of s. By induction hypothesis, we have $E(s, h(t')) \subseteq T(t')$ so $v \in T(t')$.

Case 2.2: $h(t') > h(\sigma_{t_0})$.

So $1 + h(\sigma_{t_0}) \leq h(t') \leq h(t) - 1 = h(t_0) + h(\sigma_{t_0}) - 1$.
Hence $h(t_0) = 2$ and $h(t') = h(t) - 1$.
Consequently, there exists $t'_0 \in T(t_0)$ such that $t' = \sigma(t'_0)$ and $h(t'_0) = 1$.
As t_0 is entire and normed, $x \in V(t_0) = V(t'_0)$. Hence $v = \sigma(x) \in T(t')$.

In all cases $v \in T(t')$.
□

We are now in a position to show that every accessible prefix rewriting graph of an entire system is effectively a regular graph.

Theorem B.10 *Any accessible prefix transition graph of any entire term rewriting system is a regular graph.*

Let R be a labelled entire system and r be a term.
We will construct a deterministic graph grammar G and a hyperarc H such that $R.T(F)^*/r$ is isomorphic to the regular graph generated by G from H.
From Proposition B.2, we can suppose that R is entire, normed, proper, of height $h(R)$ at most 2, and that r is a constant. We denote by $P(R,r) \;=\; R.T(F)^*/r$ the prefix transition graph of R and accessible from r. As $V(r) = \emptyset$, every vertex s of $P(R,r)$ is ground, *i.e.* $V(s) = \emptyset$ for $r \underset{R}{\longmapsto}^* s$. The deterministic graph grammar G we are going to construct, generates by parallel rewriting the graph

$$P(R,r) \quad = \quad P(R,r)_0 \cup P(R,r)_1 \cup \ldots$$

by slices

$$P(R,r)_n = \{\ (s \overset{a}{\longrightarrow} t) \ \in\ P(R,r) \mid h(s) = n\ \}$$

of arcs whose sources are of growing height. In $n+1$ parallel rewritings, the grammar G will generate the graph $P(R,r)_0 \cup \ldots \cup P(R,r)_n$ plus a set of non-terminal hyperarcs, whose vertices of $P(R,r)$ of height $n+1$ of a non-terminal hyperarc are joined by the relation

$$P \quad = \quad \{\ (s,t) \mid r \underset{R}{\longmapsto}^* s \ \wedge\ r \underset{R}{\longmapsto}^* t \ \wedge\ h(s) = h(t) \ \wedge\ E(s,h(s)-2) = E(t,h(s)-2)\ \}$$

where by extension $E(s,n) = \emptyset$ for $n < 0$.
Note that P is an equivalence and from Proposition B.6, P is decidable. Denote by $\underset{R}{\longmapsto}_n$ the prefix rewriting by vertices of height at least $n \geq 0$, *i.e.*

$$s \underset{R}{\longmapsto}_n t \quad if \quad s \underset{R}{\longmapsto} t \ \wedge\ h(s) \ \geq\ n \ \wedge\ h(t) \ \geq\ n.$$

From a non-terminal hyperarc $K(s)$ associated to a vertex s, the grammar G will generate the graph $Q(R,P(s))$ where for every set T of terms on $F \cup V$,

$$Q(R,T) \quad = \quad \{\ u \overset{a}{\longrightarrow} v \mid \exists\, t \in T,\ t\ (\underset{R}{\longmapsto}_{h(t)})^* \ u \ \wedge\ u \overset{a}{\underset{R}{\longmapsto}} v\ \}.$$

The non-terminal hyperarc $K(s)$ will join the set $Q(P(s))$ of the vertices of $Q(R,P(s))$ of height $\leq h(s)$, *i.e.* for every vertex t,

$$Q(t) \quad = \quad \{\ u \mid t\ (\underset{R}{\longmapsto}_{h(t)})^* \circ \underset{R}{\longmapsto} u \ \wedge\ h(u) \ \leq\ h(t)\ \} \cup \{t\}.$$

i) We want to define an equivalence relation $\equiv$ on the set of the vertices of $P(R,r)$, which is decidable, of finite index and finer than the equivalence of pairs (s,t) such that $Q(R,P(s))$ is isomorphic to $Q(R,P(t))$. For this and to every vertex s of $P(R,r)$, we associate an injection

$$I_s \ :\ E(s,h(s)-2) \ \hookrightarrow\ V.$$

The inverse relation $\sigma_s \;=\; I_s^{-1}$ is a substitution. Let us complete I_s into the relation

$$J_s \quad = \quad \{\ (t,x) \mid x \in Im(I_s) \ \wedge\ V(t) \ \subset\ Im(I_s) \ \wedge\ x \neq t \ \wedge\ \sigma_s(x) = \sigma_s(t)\ \}$$

in such a way that the suffix rewriting according to J_s is *canonical* (*i.e.* of finite termination and confluent. Let $t{\downarrow}J_s$ be the *normal form* of a term t according to J_s, *i.e.* $t{\downarrow}J_s$ is the minimal term for the instantiation such that $\sigma_s(t{\downarrow}J_s) = t$. Consider the equivalence relation $\equiv$ on the vertices of $P(R,r)$, defined by

$$s \equiv t \quad \text{if there exists a substitution } j \text{ from } V \text{ into } V \text{ s.t. } Q(P(s)){\downarrow}J_s = j(Q(P(t)){\downarrow}J_t).$$

The number of variables in $Q(P(s)){\downarrow}J_s$ is equal to the cardinality of $E(s,h(s)-2)$ which is from Lemma B.9, at most equal to the maximal arity of the functions in R. Furthermore, if $h(s) \leq 3$ then $J_s = \emptyset$ else $h(t{\downarrow}J_s) = 3$. Thus $\equiv$ is of finite index. We will establish (point iv) that $Q(R,P(s))$ is isomorphic to $Q(R,P(t))$ for $s \equiv t$, and we will show (point v) that $\equiv$ is decidable. First, we need to show (point ii and point iii) some intermediate results.

ii) Let $s\ P\ t$ be such that $h(s) \geq 4$ and let us show that $t{\downarrow}J_s$ is normed and entire.

a) Let us prove that $t{\downarrow}J_s$ is normed.

Let u be a (proper and) ground subterm of $t{\downarrow}J_s$. As $h(s) \geq 4$ and u is irreducible for the suffix rewriting according to I_s, we have $h(u) < h(s) - 2$. Furthermore $\sigma_s(t{\downarrow}J_s) = t$ and $h(t) = h(s)$. Thus u is a subterm of a subterm v of $t{\downarrow}J_s$ of maximal height such that $h(\sigma_s(v)) < h(s) - 2$. As v is irreducible according to J_s, we have $\sigma_s(v) \notin E(t, h(s) - 2)$. Therefore $\sigma_s(v) \in F_0$ so $h(v) = 0$ hence $h(u) = 0$. Finally $t{\downarrow}J_s$ is normed.

b) Let us prove that $t{\downarrow}J_s$ is entire.

Let u be a subterm of $t{\downarrow}J_s$ of height $h(u) \neq 0$. Similarly to (a), we have $h(\sigma_s(u)) \geq h(s) - 2$. So $\sigma_s(u)$ has a subterm v of height $h(s) - 2$. So v is a subterm of $\sigma_s(t{\downarrow}J_s) = t$ and $v{\downarrow}J_s$ is a subterm of u. By definition of P and from Lemma B.9, we have $E(s, h(s) - 2) = E(t, h(s) - 2) = T(v)$. So, every variable of $t{\downarrow}J_s$ is a variable of $v{\downarrow}J_s$, hence of u. Thus $V(u) = V(t{\downarrow}J_s)$ hence $t{\downarrow}J_s$ is entire.

iii) Let us show that $Q(R,P(s)) = \sigma_s(Q(R,P(s){\downarrow}J_s))$.

If $h(s) \leq 3$ then $J_s = \emptyset$ hence the equality.

a) Let us prove that $\sigma_s(Q(R,P(s){\downarrow}J_s)) \subseteq Q(R,P(s))$ for $h(s) \geq 4$.

Let $u \stackrel{a}{\longrightarrow} v$ an arc of $Q(R,P(s){\downarrow}J_s)$. There exists a sequence $t_0,\ldots,t_n$ such that $t_0 \in P(s){\downarrow}J_s$, $t_n = u$, $t_i \underset{R}{\longmapsto}_{h(t_0)} t_{i+1}$ for every $0 \leq i < n$.

As $h(s) \geq 4$, $h(t_0) = 3$, $V(t_0) \neq \emptyset$ and from (ii), t_0 is entire and normed. From Lemma B.7, Lemma B.4 , Lemma B.3, and by induction on $0 \leq i \leq n$, we have

$$V(t_i) = V(t_0),\ t_i \text{ is normed and entire.}$$

Furthermore $h(t_i) \geq h(t_0)$ and from Lemma B.5, we obtain

$$h(\sigma_s(t_i)) \geq h(\sigma_s(t_0)) = h(s).$$

In consequence $\sigma_s(t_i) \underset{R}{\longmapsto}_{h(s)} \sigma_s(t_{i+1})$ for every $0 \leq i < n$.

Furthermore $\sigma_s(u) \underset{R}{\stackrel{a}{\longmapsto}} \sigma_s(v)$ hence $\sigma_s(u) \stackrel{a}{\longrightarrow} \sigma_s(v) \in Q(R,P(s))$.

b) Let us prove that $Q(R,P(s)) \subseteq \sigma_s(Q(R,P(s){\downarrow}J_s))$ for $h(s) \geq 4$.

Let $u \stackrel{a}{\longrightarrow} v \in Q(R,P(s))$. There exists a sequence $t_0,\ldots,t_n$ such that $t_0 \in P(s)$, $t_n = u$, $t_i \underset{R}{\longmapsto}_{h(s)} t_{i+1}$ for every $0 \leq i < n$.

Let us set $s_0 = t_0{\downarrow}J_s$. From (ii), s_0 is entire and normed. Furthermore $\sigma_s(s_0) = t_0$, $V(s_0) \neq \emptyset$ and $h(s_0) = 3$. By induction on $1 \leq i \leq n$, we construct an entire and normed term s_i such that

$$s_{i-1} \underset{R}{\longmapsto} s_i \;,\; \sigma_s(s_i) = t_i \text{ and } h(s_i) \geq 3.$$

In fact, suppose that s_{i-1} is entire and normed, $V(s_{i-1}) \neq \emptyset, \sigma_s(s_{i-1}) = t_{i-1}$ and $h(s_{i-1}) \geq 3$. Then, there exists a term s_i such that $s_{i-1} \underset{R}{\longmapsto} s_i$ and $\sigma_s(s_i) = t_i$. From Lemma B.7, Lemma B.4 and Lemma B.3, $V(s_i) \neq \emptyset$, s_i is entire and normed. At last and from Lemma B.5, we have

$$\begin{array}{lrcl} & h(s) & \leq & h(t_i) = h(\sigma_s(s_i)) = h(s_i) + h((\sigma_s)_{s_i}) \\ \text{and} & h((\sigma_s)_{s_i}) & \leq & h((\sigma_s)_{s_0}) = h(\sigma_s)(s_0)) - h(s_0) = h(s) - 3, \\ \text{hence} & h(s) & \leq & h(s_i) + h(s) - 3 \;\; \textit{i.e. } h(s_i) \geq 3. \end{array}$$

Thus $s_0 \; (\underset{R}{\longmapsto}_{h(s_0)})^* \; s_n$. As $h(s_n) \neq 0$, there exists s_{n+1} such that $s_n \overset{a}{\underset{R}{\longmapsto}} s_{n+1}$ and $\sigma_s(s_{n+1}) = v$. Finally $(s_n \overset{a}{\longrightarrow} s_{n+1}) \in Q(R, P(s){\downarrow}J_s)$ with $\sigma_s(s_n) = u$ and $\sigma_s(s_{n+1}) = v$.

iv) Let us show that $s \equiv t$ implies that $Q(R, P(s))$ is isomorphic to $Q(R, P(t))$.
For every term t, $\sigma_s(t{\downarrow}J_s) = \sigma_s(t)$. Therefore and from (iii), we have

$$\begin{array}{lrcl} & Q(R, P(s)) & = & \sigma_s(Q(R, P(s){\downarrow}J_s)) \; = \; \sigma_s(Q(R, P(s){\downarrow}J_s){\downarrow}J_s), \\ \textit{i.e.} & Q(R, P(s)) & = & \{\; \sigma_s(u{\downarrow}J_s) \overset{a}{\longrightarrow} \sigma_s(v{\downarrow}J_s) \mid u \overset{a}{\longrightarrow} v \;\; \in Q(R, P(s){\downarrow}J_s) \;\}. \end{array}$$

a) Let us prove that σ_s is injective on $Q(R, P(s){\downarrow}J_s){\downarrow}J_s$.
Let u and v be vertices of $Q(R, P(s){\downarrow}J_s){\downarrow}J_s$ such that $\sigma_s(u) = \sigma_s(v)$. Suppose that $u \neq v$. By symmetry of u and v, we can suppose there exist a variable x of u and a subterm v' of v such that $\sigma_s(x) = \sigma_s(v')$ and $x \neq v'$. Thus $(v', x) \in J_s$ so v is reducible for the suffix rewriting according to J_s, which is impossible. Thus $Q(R, P(s))$ is isomorphic to $Q(R, P(s){\downarrow}J_s){\downarrow}J_s$.

b) Let us prove that if $t \in P(s){\downarrow}J_s$ and $t \; (\underset{R}{\longmapsto}_{h(t)})^* \; u$ then $u{\downarrow}J_s = u$ and $V(u) = \emptyset = V(t)$.
If $h(s) \leq 3$ then $J_s = \emptyset$ hence $u{\downarrow}J_s = u$ and $V(u) = V(t)$.
If $h(s) \geq 4$ then we have seen in point (iii) (a) that $V(u) = V(t)$ and, u is entire and normed. Let v be a subterm of u of height $h(v) \geq 1$. As u is normed, $V(v) \neq \emptyset$, and as u is entire, $V(v) = V(u) = V(t)$. Thus and for every variable x such that $\sigma_s(x) \neq x$, $x \in V(v)$ hence $h(\sigma_s(v)) \geq 1 + h(\sigma_s(x))$, therefore $\sigma_s(v) \neq \sigma_s(x)$ *i.e.* $(v, x) \notin J_s$. Thus $u{\downarrow}J_s = u$.

c) Let us prove that the set of terms in $Q(P(s))$ of height $h(s)$ is equal to $P(s)$.
By definition of Q, $P(s) \subseteq Q(P(s))$. Conversely, let $t \in Q(P(s))$ of height $h(t) = h(s)$ and prove that $t \in P(s)$.
If $h(s) \leq 3$ then $Q(P(s))$ [resp. $P(s)$] is the set of vertices of $P(R,r)$ of height $\leq h(s)$ [resp. $= h(s)$], hence $t \in P(s)$.
Suppose that $h(s) > 3$. For every vertex u of $Q(R, P(s){\downarrow}J_s)$, let us prove that

$$\begin{array}{lrcll} & h(\sigma_s(u)) & = & h(u) + h(s) - 3 & \text{if } \; h(u) \geq 3 \\ \text{and} & h(\sigma_s(u)) & < & h(s) & \text{if } \; h(u) < 3. \end{array}$$

In fact, if $h(u) \geq 3$, we have seen that u is entire and normed, $V(u) = V(t)$ and from Lemma B.5, we have $h(\sigma_s(u)) = h(u) + h(\sigma_s) = h(u) + h(s) - 3$.
If $h(u) < 3$ then $h(\sigma_s(u)) \leq h(u) + h(\sigma_s) < 3 + h(s) - 3 = h(s)$.
From (iii), there exists a vertex u of $Q(R, P(s){\downarrow}J_s)$ such that $t = \sigma_s(u)$. Thus $h(u) = 3$ and from (iii) (a), u is entire and normed, and $V(u) = \sigma_s(E(s, h(s) - 2))$. There is a subterm v of u of height $h(v) = 1$. As u is normed and entire, $V(v) = V(u) = \sigma_s(E(s, h(s) - 2))$. Thus $T(\sigma_s(v)) = E(s, h(s) - 2)$. From Lemma B.9,

$T(\sigma_s(v)) = E(t, h(\sigma_s(v))) = E(t, h(s)-2)$. Finally $s\ P\ t$, *i.e.* $t \in P(s)$. Thus

$$P(s) \quad = \quad \{\ t \in Q(P(s)) \mid h(t) = h(s)\ \}.$$

In particular, for every $t \in Q(P(s))$, $h(t) = h(s)$ iff $h(t{\downarrow}J_s) = min\{h(s), 3\}$. Therefore

$$P(s){\downarrow}J_s \quad = \quad \{\ t{\downarrow}J_s \mid t \in Q(P(s)) \quad \wedge \quad h(t{\downarrow}J_s) = min\{h(s),3\}\ \}\ . \tag{1}$$

d) Suppose that $s \equiv t$ and show that $Q(R, P(s))$ is isomorphic to $Q(R, P(t))$.
There exists a bijection j from V to V such that $Q(P(s)){\downarrow}J_s = j(Q(P(t)){\downarrow}J_t)$.

$$\begin{aligned} \text{Therefore } j(Q(R, P(t){\downarrow}J_t){\downarrow}J_t) &= Q(R, j(P(t){\downarrow}J_t)){\downarrow}j(J_t) \\ &= Q(R, P(s){\downarrow}J_s){\downarrow}j(J_t) && \text{from (1)} \\ &= Q(R, P(s){\downarrow}J_s){\downarrow}J_s && \text{from (b).} \end{aligned}$$

Thus $Q(R, P(s){\downarrow}J_s){\downarrow}J_s$ is isomorphic to $Q(R, P(t){\downarrow}J_t){\downarrow}J_t$ and from (a), $Q(R, P(s))$ is isomorphic to $Q(R, P(t))$.

v) Let us show that $\equiv$ is decidable, *i.e.* $Q(P(s)){\downarrow}J_s$ is constructible.
If $h(s) \leq 3$ then $J_s = \emptyset$ and $Q(P(s))$ is the set of vertices of $P(R, r)$ of height $\leq h(s)$.
Suppose that $h(s) > 3$. From (iv) (c), $Q(P(s)) = \sigma_s(Q(P(s){\downarrow}J_s))$ hence

$$Q(P(s)){\downarrow}J_s \quad = \quad Q(P(s){\downarrow}J_s){\downarrow}J_s$$

Let us prove that $Q(P(s){\downarrow}J_s)$ is constructible. As $P(s){\downarrow}J_s$ is constructible and from (iv)(c), it remains to determine the set $Q(P(s){\downarrow}J_s)$ restricted to the terms of height ≤ 2, *i.e.* the set

$$A(s) \quad = \quad \{\ u \mid \exists\, t \in P(s){\downarrow}J_s,\ t\ (\underset{R}{\longmapsto}{}_3)^* \circ \underset{R}{\longmapsto}\ u \quad \wedge \quad h(u)\ \leq\ 2\ \}.$$

Let $t \in P(s){\downarrow}J_s$. Take a constant $\$ \in F_0 - F(R)$ not in R, and denote by $u_\$$ the term obtained from a term u by replacing by $\$$ every immediate subterm of u whose height ≥ 3. If $t\ (\underset{R}{\longmapsto}{}_3)^*\ u$ then $u_\$$ belongs to the following set:

$$\begin{aligned} T \quad = \quad \{\ & u \in T(F(R) \cup V(t) \cup \{\$\}) \mid u \text{ is entire and normed } \wedge \\ & \forall v \in T(u) - \{\$\},\ h(v) \leq 2\ \wedge\ \$ \notin F(v)\ \}. \end{aligned}$$

Let b be the cardinality of T and by induction on $n \geq 0$, let us prove the following property:

$$\begin{aligned} t \in P(s){\downarrow}J_s \quad \wedge \quad t\ (\underset{R}{\longmapsto}{}_3)^n \circ \underset{R}{\longmapsto}\ u \quad \wedge \quad h(u) \leq 2 \quad \Longrightarrow \\ \exists t' \in P(s){\downarrow}J_s\ ,\ \exists q \leq b\ ,\ t'\ (\underset{R}{\longmapsto}{}_3)^q \circ \underset{R}{\longmapsto}\ u\ . \end{aligned} \tag{2}$$

If $n \leq b$ then $t' = t$ and $q = n$ satisfy property (2).
Suppose that $n > b$.

there exists a sequence $t_0, \ldots, t_n$ such that $t_0 = t$, $t_n \underset{R}{\longmapsto} u$, $t_i \underset{R}{\longmapsto}{}_3\, t_{i+1}$ for every $0 \leq i < n$. Let $m = max\{\ i \mid h(t_i) = 3\ \}$. From (iv)(c), $t_m \in P(s){\downarrow}J_s$. If $m > 0$ then (1) is true by induction hypothesis.
Suppose that $m = 0$. As $n > b$, there exists $0 < i < j \leq n$ such that $(t_i)_\$ = (t_j)_\$$. Thus $t_i\ (\underset{R}{\longmapsto}{}_3)^{n\text{-}j} \circ \underset{R}{\longmapsto}\ u$ because $h(u) \leq 2$ hence the immediate subterms of t_j of height

≥ 3 are irrelevant for getting u.

Thus $t\ (\underset{R}{\longmapsto}{}_3)^{n\text{-}j+i} \circ \underset{R}{\longmapsto} u$ hence we obtain (2) by induction hypothesis.

This ends the induction, and property (2) is established. Consequently

$$A(s) \quad = \quad \{\ u \mid \exists\ t \in P(s){\downarrow}J_s,\ \exists\ n \leq b,\ t\ (\underset{R}{\longmapsto}{}_3)^n \circ \underset{R}{\longmapsto} u \quad \wedge \quad h(u)\ \leq\ 2\ \}$$

is constructible, hence $Q(P(s){\downarrow}J_s) = P(s){\downarrow}J_s \cup A(s)$ is constructible. Therefore $Q(P(s)){\downarrow}J_s \quad = \quad Q(P(s){\downarrow}J_s){\downarrow}J_s$ is constructible, hence $\equiv$ is decidable.

vi) We can now give a (non deterministic) procedure for the construction of a deterministic graph grammar G generating $P(R,r)$.

Given a vertex s, we denote by $K(s)$ the set of hyperarcs labelled by s whose vertices are the terms of $Q(P(s))$, and which are distincts, *i.e.*

$$K(s) \quad = \quad \{ss_1 \ldots s_p \mid \{s_1,\ldots,s_p\} = Q(P(s)) \quad \wedge \quad 1 \leq i < j \leq p \quad \Rightarrow \quad s_i \neq s_j\ \}.$$

Initially G has only one rule

$$G \quad = \quad \{(H_0,\emptyset)\} \quad \text{where} \quad H_0 \in K(r).$$

The construction of G stops when every rule in G has a non empty right hand side. Else, given a rule $(H,\emptyset)$ of G, we are going to change (point d) the right hand side of this rule, and adding (point c) in G the non-terminal hyperarcs whose rules we want to construct. We denote by s the label of H, *i.e.* $s = H(1)$ that is $H \in K(s)$.

a) The new right hand side will have as terminal arcs, the set $T(s)$ of the arcs in $P(R,r)$ whose sources are in $P(s)$, *i.e.*

$$T(s) \quad = \quad \{\ u \stackrel{a}{\longrightarrow} v \mid u \underset{R}{\stackrel{a}{\longmapsto}} v \quad \wedge \quad u\ \in\ P(s)\ \}.$$

b) We will extract one vertex by non-terminal hyperarc to be constructed.

Let M be the set of vertices of $T(s)$, of height $h(s)+1$ and sources of arcs in $P(R,r)$, *i.e.*

$$M \quad = \quad \{\ v \mid \exists u \in\ P(s),\ u \underset{R}{\longmapsto} v \quad \wedge \quad h(v) = h(s)+1 \quad \wedge \quad \exists t,\ v \underset{R}{\longmapsto} t\ \}.$$

Let us take a set A of representatives of the quotient of M by P, *i.e.* A is a minimal subset of M such that

$$\bigcup\{\ P(t) \mid t \in A\ \} \quad = \quad \bigcup\{\ P(t) \mid t \in M\ \}.$$

From (iv)(c), remark that for $s\ (\underset{R}{\longmapsto}{}_{h(s)})^* \ t$ and $h(s) = h(t)$, we have $s\ P\ t$.

c) To every $t \in A$, we will associate a non-terminal hyperarc $N(t)$.

If t is equivalent to a non-terminal of G, *i.e.* $\exists\ ut_1 \ldots t_q \in Dom(G),\ u\ \equiv\ t$, then there is a bijection j from V to V such that $j(Q(P(u)){\downarrow}J_u) \quad = \quad Q(P(t)){\downarrow}J_t$, and we define

$$N(t) \quad = \quad u\sigma_t(j(t_1{\downarrow}J_u))\ldots\sigma_t(j(t_q{\downarrow}J_u)).$$

Otherwise, we choose $N(t)$ in $K(t)$ that we add as a non-terminal hyperarc of G, *i.e.*

$$G \quad = \quad G \cup \{(N(t),\emptyset)\}.$$

d) $$G = (G - (H, \emptyset)) \cup \{ (H, T(s) \cup \{ N(t) \mid t \in A \}) \}.$$

By construction and for every $H \in Dom(G) \cap K(s)$, $G^{\omega}(H)$ is isomorphic to $Q(R, P(s))$. In particular when $s = r$, $G^{\omega}(H_0)$ is isomorphic to $P(R, r)$.
□