Word-mappings of level 2

Julien Ferté, Nathalie Marin and Géraud Sénizergues [1]

LIF, Université d'Aix-Marseille 1 and institution of N. Marin and LaBRI, Université Bordeaux 1, FRANCE

Abstract

A sequence of natural numbers is said to have *level* k, for some natural integer k, if it can be computed by a deterministic pushdown automaton of level k ([Fratani-Sénizergues, APAL, 2006]). We show here that the sequences of level 2 are exactly the rational formal power series over one undeterminate. More generally, we study mappings *from words to words* and show that the following classes coincide:

- the mappings which are computable by deterministic pushdown automata of level 2
- the mappings which are solution of a system of catenative recurrence equations
- the mappings which are definable as a Lindenmayer system of type $HDT0L$.

We illustrate the usefulness of this characterization by proving three statements about formal power series, rational sets of homomorphisms and equations in words.

Key words: :Iterated pushdown automata, L-systems, integer sequences, rational power series, word sequences, word equations.
1991 MSC: 68Q45, 11B85, 40B05, 20M05

Version: 22 February 2012

Contents

Email address:
julien.ferte@lif.univ-mrs.fr,nathalie.marin@gmail.com,ges@labri.u-bordeaux.fr (Julien Ferté, Nathalie Marin and Géraud Sénizergues).

[1] Work supported by the project: ANR 2010 BLAN 0202 01 FREC.

1 Introduction

A sequence of natural numbers (i.e. a mapping from $\mathbb{N}$ to $\mathbb{N}$) is said to have *level k*, for some natural integer k, if it can be computed by a deterministic pushdown automaton of level k ([FS06]). We concentrate here on the level 2 of this hierarchy of sequences. We show that the sequences of level 2 are exactly the $\mathbb{N}$-rational formal power series over one undeterminate. More generally, we study mappings *from words to words* and show that the following classes coincide:
- the mappings which are computable by deterministic pushdown automata of level 2
- the mappings which are solution of a system of catenative recurrence equations
- the mappings which are definable as a Lindenmayer system of type $HDT0L$.

Applying this characterization, we:
- give a new proof of the skimming theorem ([Sch70]) and give a natural extension of this theorem to mappings from words to words
- prove that the set of rational sets of endomorphisms of a free monoid is not closed under intersection
- show that the set of solutions of a finite set of quadratic equations (over words) is an indexed language (i.e. is recognized by a pda of level 2).

2 Preliminaries

2.1 Sets

Given a set E, we denote by $\mathcal{P}(E)$ the set of its subsets and by $\mathcal{P}_f(E)$ the set of its *finite* subsets.
A *binary relation* from a set E into a set F is a subset R of $E \times F$. The domain and image of R are defined by:

$$\mathrm{dom}(R) := \{x \in E \mid \exists y \in F, (x, y) \in R\}, \;\; \mathrm{im}(R) := \{y \in F \mid \exists x \in E, (x, y) \in R\}.$$

We denote by $\circ$ the composition of binary relations: if $R \subseteq E \times F, R' \subseteq F \times G$ then:

$$R \circ R' := \{(x, z) \in E \times G \mid \exists y \in F, (x, y) \in R \;\wedge\; (y, z) \in R'\}$$

A *function* from the set E into the set F is a binary relation $f \subseteq E \times F$ such that,

$$\forall (x, y), \forall (x', y') \in f, x = x' \Rightarrow y = y'$$

Note that, when using a functional notation, we still use the composition operator $\circ$ as above i.e.

$$(f \circ g)(x) := g(f(x)).$$

We call *mapping* (or simply, *map*) from E to F any function $f : E \to F$ such that $\mathrm{dom}(f) = E$.

2.2 Abstract rewriting

Let E be some set and $\to \subseteq E \times E$. We note $\to^0 := \mathrm{Id}_E, \to^1 := \to$, for every $n \geq 1$ $\to^{n+1} := \to \circ \to^n$ and finally:

$$\to^* := \bigcup_{n=0}^{\infty} \to^n .$$

The relation $\to^*$ is the reflexive and transitive closure of the binary relation $\to$. The notation $e \to^\infty$ means that there exists some infinite sequence

$$e_0, e_1, \cdots e_i, e_{i+1}, \ldots, e_n, \ldots$$

such that $e = e_0$ and, for every $i \in \mathbb{N}$, $e_i \to e_{i+1}$. The relation $\to$ is called *noetherian* iff, $\forall e \in E, \neg(e \to^\infty)$. The relation $\to$ is called *confluent* iff, for every $e, e_1, e_2 \in E$

$$(e \to^* e_1 \wedge e \to^* e_2) \Rightarrow (\exists e' \in E, e_1 \to^* e' \wedge e_2 \to^* e').$$

The relation $\to$ is called *strongly locally confluent* iff, for every $e, e_1, e_2 \in E$

$$(e \to e_1 \wedge e \to e_2) \Rightarrow (\exists e' \in E, \exists n_1, n_2 \in \{0,1\}, e_1 \to^{n_1} e' \wedge e_2 \to^{n_2} e').$$

Lemma 1 ([Hue80]) *If $\to$ is strongly locally confluent, then $\to$ is confluent.*

An element $e \in E$ is called a *irreducible* (w.r.t. $\to$) iff, there exists no $e' \in E$ such that $e \to e'$. We denote by $\mathrm{IRR}(\to)$ the set of irreducible elements of E (w.r.t. $\to$).

Lemma 2 ([Hue80]) *If $\to$ is confluent and noetherian, for every $e \in E$ there exists a unique irreducible element $e' \in E$ such that $e \to^* e'$.*

Under the hypothesis of Lemma 2 we call *normal form* of e the unique irreducible $e' \in E$ such that $e \to^* e'$. We denote by $\rho_\to : E \to \mathrm{IRR}(\to)$ the map defined by:

$$\forall e \in E, e \to^* \rho_\to(e) \text{ and } \rho_\to(e) \in \mathrm{IRR}(\to).$$

2.3 Monoids

We recall a monoid is a triple $\langle M, \cdot, \mathbf{1}\rangle$ such that, M is a set (the carrier of the monoid), $\cdot$ is a composition law which is associative and $\mathbf{1}$ is a neutral element for $\cdot$ (on both sides). An equivalence relation $\sim$ over M is called a *right-regular* equivalence if and only if , for every $x, y, z \in M$

$$x \sim y \Rightarrow x \cdot z \sim y \cdot z.$$

(the notion of left-regular equivalence is defined analogously).
Given two monoids $\mathbb{M}_1 := \langle M_1, \cdot, \mathbf{1}_1\rangle$, $\mathbb{M}_2 := \langle M_2, \cdot, \mathbf{1}_2\rangle$ a monoid-homomorphism from $\mathbb{M}_1$ to $\mathbb{M}_2$ is a map $h : M_1 \to M_2$ fulfilling: for every $x, y \in M_1$

$$h(x \cdot y) = h(x) \cdot h(y) \text{ and } h(\mathbf{1}_1) = \mathbf{1}_2.$$

We denote by $\mathrm{HOM}(\mathbb{M}_1, \mathbb{M}_2)$ the set of all monoid-homomorphism from $\mathbb{M}_1$ to $\mathbb{M}_2$. For every monoid $\mathbb{M}$, the set $\mathrm{HOM}(\mathbb{M}, \mathbb{M})$, endowed with the composition law $\circ$ and the identity map Id_M is a monoid.
Given a set X (that we see as an "alphabet"), we denote by X^* the set of all finite words labelled on this set X. We denote by $\cdot$ the concatenation law over X^* and denote by ε the empty word. The structure $\langle X^*, \cdot, \varepsilon \rangle$ is the *free monoid* over the alphabet X. Given a word $u \in X$:

$$u = x_0 x_1 \cdots x_{\ell-1}$$

we denote by $\tilde{u}$ the *mirror image* of u, which is the word:

$$\tilde{u} := x_{\ell-1} \cdots x_1 x_0.$$

Given an alphabet X, a *semi-Thue system* over X is a susbet

$$R \subseteq X^* \times X^*.$$

The rewriting relation $\rightarrow_R$ associated with R is defined by

$$(u \rightarrow_R v) \Leftrightarrow (\exists \alpha, \beta \in X^*, \exists (w, w') \in R, u = \alpha w \beta \wedge v = \alpha w' \beta).$$

All the notions recalled in the subsection 2.2 are applicable to this binary relation $\rightarrow_R$. We refer to [BO93] for a thorough exposition of this subject.

2.4 Automata

The notion of finite automaton over an input alphabet X is defined as usual. We recall in next paragraph its extension to finite automata with inputs and outputs.

2.4.1 Sequential functions

Definition 3 *A finite transducer (abbreviated ft) is a sextuple*

$$\mathcal{A} := \langle X, Y, Q, Q_-, Q_+, \delta \rangle$$

such that, X, Y, Q are finite sets, $Q_- \subseteq Q, Q_+ \subseteq Q$, and δ is a finite subset of $Q \times X^* \times Y^* \times Q$.
A generalized sequential machine (abbreviated gsm) is a finite transducer $\mathcal{A}$ such that Q_- is a singleton $\{q_-\}$, $\delta \subseteq Q \times X \times Y^* \times Q$ and, for every $q, r, s \in Q, x \in X, u, v \in Y^*$

$$(q, x, u, r), (q, x, v, s) \in \delta \Rightarrow (u = v \text{ and } r = s).$$

With the above notation, X is called the input alphabet, Y is the output alphabet, Q is the set of states Q_- (resp. Q_+) is the set of initial states (resp. the set of terminal states) and δ is the set of transitions. The transducer is a gsm when its underlying finite automaton over X^* is deterministic.
The function computed by the gsm $\mathcal{A}$ is the set of all pairs $(u, v) \in X^* \times Y^*$ such that, there exists a path in $\mathcal{A}$,

$$(q_1, x_1, v_1, q_2)(q_2, x_2, v_2, q_3) \cdots (q_n, x_n, v_n, q_{n+1})$$

with $q_1 = q_-$, every (q_i, x_i, v_i, q_{i+1}) belongs to δ, $q_{n+1} \in Q_+$ and $u = \prod_{i=1}^n x_i, v = \prod_{i=1}^n v_i$.
A map (i.e. a total function) which is computed by some gsm is called a *left sequential* map.
In this case:
- one can normalize $\mathcal{A}$ in such a way that $Q_+ = Q$.
- for every $u \in X^*, q \in Q$ we denote by $q \odot u$ the state reached by $\mathcal{A}$ starting from q and reading u and by $q * u$ the word output by $\mathcal{A}$ during this computation. Note that, in this case, $(q, u) \mapsto (q \odot u)$ is a right-action of the monoid X^* over the set Q and, for every state $q \in Q$, $u \mapsto (q * u)$ is a left-sequential map.

A map $g : X^* \to Y^*$ is called a *right sequential* map iff, the map $f : u \mapsto f(\tilde{u})$ is left-sequential. Equivalently, g is right-sequential iff it is computed by some finite transducer

$$\mathcal{A} := \langle X, Y, Q, Q_-, Q_+, \delta \rangle$$

such that the underlying finite automaton over X^* is co-deterministic.
In this case:
- Q_+ has a unique element q_+ and one can normalize $\mathcal{A}$ in such a way that $Q_- = Q$
- for every $u \in X^*, q \in Q$ we denote by $u \odot q$ the state from which $\mathcal{A}$ can reach q when reading u; we denote by $u * q$ the word output by $\mathcal{A}$ during this computation. Note that, in this case, $(q, u) \mapsto (u \odot q)$ is a left-action of the monoid X^* over the set Q and, for every state $q \in Q$, $u \mapsto (u * q)$ is a right-sequential map.

Definition 4 *Let $\sim$ be a left-regular equivalence relation of finite index over X^* and let $\mathcal{C} := X^*/\sim$. We define the map $\mathcal{G}_\sim : X^* \to (X \times \mathcal{C})^*$ by: $\forall u \in X^+$, if $u = u_1 u_2 u_3 \cdots u_n$, where $u_i \in X$, then*

$$\mathcal{G}_\sim(u) := (u_1, [u_2 \ldots u_n]_\sim) \cdot (u_2, [u_3 \ldots u_n]_\sim) \cdot \ldots \cdot (u_n, [\epsilon]_\sim)$$

and

$$\mathcal{G}_\sim(\epsilon) = (\epsilon, [\epsilon]_\sim)$$

It can be easily checked that this map is right-sequential.

2.4.2 Pushdown automata

A general notion of *pushdown automaton of level k*, for every natural integer $k \geq 1$, has been defined in ([Mas74],[Mas76]). We shall use here a slight variant of this notion, which was introduced in [DG86] and appeared particularly suitable for defining integer sequences of level k (see [FS06]). Since this paper focuses on level 2, we restrict our definitions to this level. We recall that a pushdown store of level 1 over an alphabet Γ is just a word over the alphabet Γ.

Definition 5 *Let Γ be a finite set of symbols. The set of pushdown stores of level 2 over Γ is:*

$$2 - \mathrm{pds}(\Gamma) := (\Gamma[\Gamma^*])^*$$

Examples: Let $\Gamma := \{A_0, A_1, x, y, a, b\}$ Here are some elements of $2 - \mathrm{pds}(\Gamma)$:

$$A_0[xyy]A_1[xyy]A_1[\varepsilon]A_0[aa], \quad a[\varepsilon]b[xA_1A_0]$$

A variant of this notion can be defined for two alphabets Γ_1, Γ_2:

$$2 - \mathrm{pds}(\Gamma_1, \Gamma_2) := (\Gamma_1[\Gamma_2^*])^*$$

Examples: Let $\Gamma_: = \{A_0, A_1\}, \Gamma_2 := \{x, y, a, b\}$. Here are some elements of $2 - \mathrm{pds}(\Gamma_1, \Gamma_2)$:

$$A_0[xyy]A_1[xyy]A_1[\varepsilon]A_0[aa], \quad A_0[\varepsilon]A_1[xax]$$

Let us remark that $2 - \mathrm{pds}(\Gamma)$, endowed with the concatenation product, is a monoid.

Definition 6 *Let ω be in $2 - \mathrm{pds}(\Gamma)$, ω is said* atomic *if :*

$$\forall \omega_1, \omega_2, \omega = \omega_1 \cdot \omega_2 \Rightarrow (\omega_1 = \epsilon \vee \omega_2 = \epsilon)$$

In this case, the pushdown store ω is also called an atom.

Note that the monoid $2 - \mathrm{pds}(\Gamma)$ is freely generated by its set of atoms.

Lemma 7 *Let ω be in $2 - \mathrm{pds}(\Gamma)$, $\omega \neq \epsilon$. Then :*

$$\exists! A \in \Gamma, \exists! \alpha \in \Gamma^*, \exists! \beta \in 2 - \mathrm{pds}(\Gamma), \omega = A[\alpha] \cdot \beta.$$

Example: For $\omega := A_0[xyy]A_1[xyy]A_1[\varepsilon]$ we have $A = A_0, \alpha = xyy, \beta = A_1[xyy]A_1[\varepsilon]$.
This lemma will be useful for defining some operations over $2-\text{pds}$.

Definition 8 (Reading) *Let* $\text{topsyms} : 2 - pds(\Gamma) \to \Gamma^*$ *be the map defined by:* $\forall A, B \in \Gamma, \forall \alpha \in \Gamma^*, \forall \beta \in 2 - \text{pds}(\Gamma)$

$$\begin{aligned}
\text{topsyms}(\varepsilon) &:= \varepsilon \\
\text{topsyms}(A[\varepsilon] \cdot \beta) &:= A \\
\text{topsyms}(A[B \cdot \alpha] \cdot \beta) &:= AB
\end{aligned}$$

Examples: For $\omega := A_0[xyy]A_1[xyy]A_1[\varepsilon], \omega' := A_0[\varepsilon]A_1[xyy]A_1[xx]$ we have $\text{topsyms}(\omega) = A_0x, \text{topsyms}(\omega') = A_0$.

Definition 9 (Erasing) *Let j be in $\{1,2\}$. The functions* $\text{pop}_j : 2-\text{pds}(\Gamma) \to 2-\text{pds}(\Gamma)$ *are defined by : for every* $A, B \in \Gamma, \alpha \in 1-\text{pds}(\Gamma), \beta \in 2-\text{pds}(\Gamma)$*:*

$$\begin{aligned}
\text{pop}_1(A[\alpha]\beta) &:= \beta \\
\text{pop}_1(\varepsilon) &:= \textit{undefined} \\
\text{pop}_2(A[B\alpha]\beta) &:= A[\alpha]\beta \\
\text{pop}_2(A[\varepsilon]\beta) &:= \textit{undefined.}
\end{aligned}$$

Examples:
$\text{pop}_1(A_0[xyy]A_1[xyy]A_1[\varepsilon]) = A_1[xyy]A_1[\varepsilon]$
$\text{pop}_2(A_0[xyy]A_1[xyy]A_1[\varepsilon]) = A_0[yy]A_1[xyy]A_1[\varepsilon]$.

Definition 10 (Writing) *Let j be in $\{1,2\}$ and $\gamma \in \Gamma^+$: $\gamma = A_1A_2\cdots A_n$ where $A_i \in \Gamma$. The functions* $\text{push}_j(\gamma) : 2-\text{pds}(\Gamma) \to 2-\text{pds}(\Gamma)$ *are defined by : for every* $\alpha \in 1-\text{pds}(\Gamma), \beta \in 2-\text{pds}(\Gamma)$*:*

$$\begin{aligned}
\text{push}_1(\gamma)(A[\alpha]\beta) &:= A_1[\alpha]A_2[\alpha]\cdots A_n[\alpha]\beta \\
\text{push}_1(\gamma)(\varepsilon) &:= A_1[\varepsilon]A_2[\varepsilon]\cdots A_n[\varepsilon] \\
\text{push}_2(\gamma)(A[B\alpha]\beta) &:= A[\gamma\alpha]\beta \\
\text{push}_2(\gamma)(A[\varepsilon]\beta) &:= A[\gamma]\beta \\
\text{push}_2(\gamma)(\varepsilon) &:= \textit{undefined.}
\end{aligned}$$

Examples:
$\text{push}_1(A_1)(A_0[xyy]A_1[xyy]A_1[\varepsilon]) = A_1[xyy]A_0[xyy]A_1[xyy]A_1[\varepsilon])$
$\text{push}_2(y)(A_0[xyy]A_1[xyy]A_1[\varepsilon]) = A_0[yyy]A_1[xyy]A_1[\varepsilon])$

$\text{push}_2(xy)(A_0[xyy]A_1[xyy]A_1[\varepsilon]) = A_0[xyyy]A_1[xyy]A_1[\varepsilon])$
$\text{push}_2(xy)(A_0[\varepsilon]A_1[xyy]A_1[\varepsilon]) = A_0[xy]A_1[xyy]A_1[\varepsilon])$.

Let POP denote the set $\{\text{pop}_j \,| j \in \{1,2\}\}$, PUSH$(\Gamma)$ denote the set $\{\text{push}_j(\gamma) \mid \gamma \in \Gamma^+, j \in \{1,2\}\}$ and TOPSYMB(Γ) denote $\Gamma \cup \Gamma^2$.

Definition 11 *A pushdown automata of level 2 (*2-pda*) is a* 6*-uple* $\mathcal{A} = (Q, X, \Gamma, \delta, q_0, A_0)$ *where :*

- Q *is a finite set of states*
- X *is a finite alphabet (called the input alphabet)*
- Γ *is a finite set of symbols (called the pushdown alphabet)*
- $\delta : Q \times X^{\leq 1} \times \text{TOPSYMB}(\Gamma) \rightarrow \mathcal{P}_f(Q \times (\text{PUSH}(\Gamma) \cup \text{POP}))$ *is the transition map, satisfying both following conditions :*
  $(q, \text{push}_j(\gamma')) \in \delta(p, \bar{a}, \gamma) \Rightarrow j \leq |\gamma| + 1$ $(q, \text{pop}_j) \in \delta(p, \bar{a}, \gamma) \Rightarrow j \leq |\gamma|$
- q_0 *is the initial state,* A_0 *is the initial symbol of the* 2*-pds (*2*-level pushdown storage)*

2.4.2.1 Computation

Given a $2-$pda $\mathcal{A}$, se define a notion of *computation* for this automaton.

Definition 12 *Let* $\mathcal{A} = (Q, X, \Gamma, \delta, q_0, A_0)$ *a* 2*-pda. We define :*

- $\text{Conf}_{\mathcal{A}} := Q \times X^* \times 2pds(\Gamma)$*, the set of configurations of* $\mathcal{A}$
- $\vdash_{\mathcal{A}} \subset \text{Conf}_{\mathcal{A}} \times \text{Conf}_{\mathcal{A}}$*, the move relation of* $\mathcal{A}$*, by:* $(p, u, \omega) \vdash_{\mathcal{A}} (q, v, \omega')$ *if and only if*

$$\exists (q, \text{op}) \in \delta(p, \bar{a}, \text{topsyms}(\omega)), \quad \bar{a}v = u \quad \wedge \quad \omega' = \text{op}(\omega)$$

- $\vdash^*_{\mathcal{A}}$ *is the reflexive and transitive closure of* $\vdash_{\mathcal{A}}$
- $\text{L}_{\mathcal{A}}$ *is the langage accepted with empty pushdown i.e.:*

$$\text{L}_{\mathcal{A}} := \{u \in X^* | \exists q \in Q, (q_0, u, A_0[\varepsilon]) \vdash^*_{\mathcal{A}} (q, \epsilon, \epsilon)\}$$

Let us precise what is meant by *determinism* for this kind of automata.

Definition 13 *Let* $\mathcal{A}$ *be a* 2-pda*,* $\mathcal{A}$ *is said deterministic if :* $\forall q \in Q, \gamma \in \text{TOPSYMB}(\Gamma), a \in X, \begin{cases} \text{Card}(\delta(q, \epsilon, \gamma)) \leq 1 \wedge \text{Card}(\delta(q, a, \gamma)) \leq 1 \\ \text{Card}(\delta(q, \epsilon, \gamma)) = 1 \Rightarrow \text{Card}(\delta(q, a, \gamma)) = 0 \end{cases}$

Definition 14 *Let* $\mathcal{A}$ *be a* 2*-pda,* $\mathcal{A}$ *will be said* strongly *deterministic if :*

$$\forall q \in Q, \forall \gamma \in \text{TOPSYMB}(\Gamma), \sum_{\bar{a} \in X^{\leq 1}} \text{Card}(\delta(q, \bar{a}, \gamma)) \leq 1$$

In words: the input letter read is entirely determined by the sequence of top symbols of the 2-pds and the state of $\mathcal{A}$.

2.4.2.2 Derivation Let us recall a notion of derivation introduced in [FS06] together with its basic properties.

Definition 15 *Let $\mathcal{A}$ be a* 2-pda. *We define the infinite alphabet $V_{\mathcal{A}} := Q \times 2 - pds(\Gamma) \times Q$ and the infinite set of productions, $P_{\mathcal{A}}$ by:*

(1) for every $q \in Q$, if $(p, \bar{a}, \omega) \vdash_{\mathcal{A}} (p', \epsilon, \omega')$, then $(p, \omega, q) \to_{\mathcal{A}} \bar{a}(p', \omega', q)$
(2) if $\omega = \omega_1\omega_2$ and $\omega_1 \neq \epsilon$ and $\omega_2 \neq \epsilon$, then, for every $r \in Q$, $(p, \omega, q) \to_{\mathcal{A}} (p, \omega_1, r)(r, \omega_2, q)$
(3) for every $p \in Q$, $(p, \varepsilon, p) \to_{\mathcal{A}} \varepsilon$

The rules (1) are called *transition* rules, the rules (2) are called *decomposition* rules and the rules (3) are called *epsilon*-rules. Let X be the terminal alphabet of $\mathcal{A}$. We define the binary relation $\to_{\mathcal{A}}$ as the smallest subset of $(V_{\mathcal{A}} \cup X)^* \times (V_{\mathcal{A}} \cup X)^*$ containing all the above rules (1)(2)(3) and compatible with left product and right product. The *derivation* generated by $\mathcal{A}$, denoted by $\to_{\mathcal{A}}^*$, is the reflexive and transitive closure of $\to_{\mathcal{A}}$. Given a word $U \in V_{\mathcal{A}}^*$, the language generated by U is defined as

$$\mathrm{L}_{\mathcal{A}}(U) := \{w \in X^* \mid U \to_{\mathcal{A}}^* w\}.$$

Example:
Let us define the 2-pda, over the terminal alphabet $Y := \{a, b\}$, $\mathcal{A} := (Q, Y, \Gamma, \delta, q_0, A_0)$ by:

$$\begin{array}{rcll}
Q := & \{q_0, q_1, q_{0,x}, q_{1,x}, q_{0,y}, q_{1,y}\} & & \\
\Gamma := & \{A_0, A_1, x, y, a, b\} & & \\
\delta(q_0, \varepsilon, A_i z) := & (q_{i,z}, \mathrm{pop}_2) & \text{for } i \in \{0,1\}, z \in \{x, y\} & \\
\delta(q_{0,x}, \varepsilon, A_0\bar{a}) := & (q_0, \mathrm{push}_1(A_0A_1)) & \text{for } \bar{a} \in \{x, y, \varepsilon\} & \\
\delta(q_{0,y}, \varepsilon, A_0\bar{a}) := & (q_0, \mathrm{push}_1(A_1)) & \text{for } \bar{a} \in \{x, y, \varepsilon\} & \\
\delta(q_{1,x}, \varepsilon, A_1\bar{a}) := & (q_0, \mathrm{push}_1(A_1A_0)) & \text{for } \bar{a} \in \{x, y, \varepsilon\} & \\
\delta(q_{1,y}, \varepsilon, A_1\bar{a}) := & (q_0, \mathrm{push}_1(A_1A_0)) & \text{for } \bar{a} \in \{x, y, \varepsilon\} & \\
\delta(q_0, a, A_0) := & (q_0, \mathrm{pop}_1) & & (1)\\
\delta(q_0, b, A_1) := & (q_0, \mathrm{pop}_1) & . & (2)
\end{array}$$

For every $u \in \{x, y\}^*$, we get the following basic derivations:

$$
\begin{aligned}
(q_0, A_0[xu], q_0) &\to^*_{\mathcal{A}} (q_0, A_0[u], q_0)(q_0, A_1[u], q_0)\\
(q_0, A_0[yu], q_0) &\to^*_{\mathcal{A}} (q_0, A_1[u], q_0)\\
(q_0, A_1[xu], q_0) &\to^*_{\mathcal{A}} (q_0, A_1[u], q_0)(q_0, A_0[u], q_0)\\
(q_0, A_1[yu], q_0) &\to^*_{\mathcal{A}} (q_0, A_1[u], q_0)(q_0, A_0[u], q_0)\\
(q_0, A_0[\varepsilon], q_0) &\to^*_{\mathcal{A}} a\\
(q_0, A_1[\varepsilon], q_0) &\to^*_{\mathcal{A}} b.
\end{aligned}
$$

Therefore we get:

$$
\begin{aligned}
(q_0, A_0[xxx], q_0) &\to^*_{\mathcal{A}} (q_0, A_0[xx], q_0)(q_0, A_1[xx], q_0)\\
&\to^*_{\mathcal{A}} (q_0, A_0[x], q_0)(q_0, A_1[x], q_0)(q_0, A_1[x], q_0)(q_0, A_0[x], q_0)\\
&\to^*_{\mathcal{A}} (q_0, A_0[\varepsilon], q_0)(q_0, A_1[\varepsilon], q_0)^2(q_0, A_0[\varepsilon], q_0)\cdot\\
&\qquad (q_0, A_1[\varepsilon], q_0)(q_0, A_0[\varepsilon], q_0)^2(q_0, A_1[\varepsilon], q_0)\\
&\to^*_{\mathcal{A}} abbabaab.
\end{aligned}
$$

Let us recall a useful lemma from [FS06].

Lemma 16 (Derivation vs computation 1) *Let $\mathcal{A}$ be a 2-pda with terminal alphabet X and pushdown alphabet Γ. Then : $\forall u \in X^*, p, q \in Q, \omega \in 2-\mathrm{pds}(\Gamma)$,*

$$(p, \omega, q) \to^*_{\mathcal{A}} u \Leftrightarrow (p, u, \omega) \vdash^*_{\mathcal{A}} (q, \epsilon, \epsilon)$$

Some more precision about the steps used in one $\to_{\mathcal{A}}$-derivation and its corresponding $\vdash_{\mathcal{A}}$-computation will be needed in the sequel. For every $i \in \{1, 2, 3\}$, let us define $\to_{\mathcal{A},i}$ as the smallest subset of $(V_{\mathcal{A}} \cup X)^* \times (V_{\mathcal{A}} \cup X)^*$ containing all the rules of type (i) and compatible with left product and right product.

Lemma 17 (Derivation vs computation 2) *Let $\mathcal{A}$ be a 2-pda with terminal alphabet X, and pushdown alphabet Γ. Then : $\forall u \in X^*, n \in \mathrm{N}, p, q \in Q, \omega \in 2-\mathrm{pds}(\Gamma)$,*

$$(p, \omega, q)(\to_{\mathcal{A},1} \cup(\to_{\mathcal{A},2} \cup \to_{\mathcal{A},3})^*)^n u \Leftrightarrow (p, u, \omega) \vdash^n_{\mathcal{A}} (q, \epsilon, \epsilon)$$

In other words, the number of *transition rules* (i.e. rules of type (1)) used in the derivation is equal to the number of transition rules used in the computation.

2.4.2.3 Substitutions Given an alphabet Γ of pushdown symbols, we introduce another alphabet $\mathbb{O} = \{\Omega, \Omega', \Omega'', \ldots, \Omega_1, \Omega_2, \ldots, \Omega_n, \ldots\}$ of *undeter-*

minates. We suppose that $\Gamma \cap \mathbb{O} = \emptyset$. The set of pushdown stores of level 2 is extended to undeterminates by allowing one undeterminate to be positionned at the bottom of the innermost pushdown:

$$2 - \mathrm{pds}_{\mathbb{O}}(\Gamma) := (\Gamma[\Gamma^*(\mathbb{O} \cup \{\varepsilon\})])^*$$

Examples:

$$\omega := A_0[xy\Omega]A_1[xyy]A_1[\varepsilon]A_0[a\Omega'], \;\; \omega' := A_0[\varepsilon]A_1[\Omega]$$

are pushdown stores with undeterminates in $\mathbb{O}$. Given $\omega \in 2 - \mathrm{pds}_{\mathbb{O}}(\Gamma)$ and a family of pairs $(\Omega_i, v_i) \in \mathbb{O} \times \Gamma^*$ (for $i \in I$) we denote by

$$[u_i/\Omega_i, i \in I] : 2 - \mathrm{pds}_{\mathbb{O}}(\Gamma) \to 2 - \mathrm{pds}(\Gamma)$$

the substitution consisting in replacing every occurrence of the letter Ω_i by the word u_i. Since the undeterminates can occur only as bottom symbols, the resulting word is also a pushdown store of level 2. With the examples above:

$$\omega[yx/\Omega, b/\Omega'] = A_0[xyyx]A_1[xyy]A_1[\varepsilon]A_0[ab], \;\; \omega'[yx/\Omega, b/\Omega'] = A_0[\varepsilon]A_1[yx].$$

Given some $2 - \mathrm{pda}\, \mathcal{A}$ over a pushdown alphabet included in Γ, we extend the relations $\to^*_{\mathcal{A}}, \vdash^*_{\mathcal{A}}$ to the pushdown alphabet $\Gamma \cup \mathbb{O}$.

Lemma 18 (Substitution principle) *Let $(v_i)_{i \in I}$ be some family of words in Γ^*, $\omega, \omega' \in 2 - \mathrm{pds}_{\mathbb{O}}(\Gamma)$. If*

$$(p, \omega, q) \to^*_{\mathcal{A}} (p', \omega', q')$$

then,

$$(p, \omega[u_i/\Omega_i, i \in I], q) \to^*_{\mathcal{A}} (p', \omega'[u_i/\Omega_i, i \in I], q').$$

The key-idea for this lemma is that, since $\Gamma \cap \mathbb{O} = \emptyset$, the symbols Ω_i can be copied or erased during the derivation, but they cannot *influence* the sequence of rules used in that derivation.

2.5 Sequences

2.5.1 2-computable sequences

Definition 19 (2-computable sequences) *A mapping $f : X^* \to Y^*$ is called a* 2-computable mapping *(or sequence) iff there exists a 2-sdpda $\mathcal{A}$, over a pushdown-alphabet $\Gamma \supseteq X$, with terminal alphabet Y, such that, for every $u \in X^*$:*

$$(q_0, f(u), A_0[u]) \vdash^*_{\mathcal{A}} (q_0, \varepsilon, \varepsilon).$$

One denotes by $\mathbb{S}_2(X^*, Y^*)$ the set of all 2-computable sequences mappings from X^* to Y^*.

The particular case where $\mathrm{Card}(A) = \mathrm{Card}(B) = 1$ was studied in [FS06]. In words: the mapping f is computed by the sdpda $\mathcal{A}$ if, starting with a 2-pds consisting of just the atom $A_0[u]$, and with the initial state q_0, it accepts (on the input tape) the word $f(u)$. Since the automaton is *strongly* deterministic, the word $f(u)$ is completely determined by the initial configuration $q_0 A_0[u]$.

Remark 20 *Changing of point of view, one could see u as the input of the computation and $f(u)$ as its output. From this point of view, the automaton $\mathcal{A}$ is (up to some technical details) a* deterministic transducer *in the class denoted by* $\mathrm{D}_t\,\mathrm{REG}(\mathrm{P}(\mathrm{WD}))$ *in [EV86], for the following storage structure* WD*:*
- its domain is Γ^*
- the predicates are $u = \varepsilon?$, $\mathrm{leftmost-letter}(u) = \gamma?$ (for all the letters $\gamma \in \Gamma$)
- the operations are $u \mapsto \gamma \cdot u$ and $\gamma \cdot u \mapsto u$ (for all the letters $\gamma \in \Gamma$)

Example: The pda of the example given after Definition 15 is strongly deterministic. Let $X := \{x, y\}$ which is included in Γ. One can check that, for every $u \in X^*$, there exists some derivation starting on $(q_0, A_0[u], q_0)$ and ending in a word $f(u) \in Y^*$. Hence there is a computation of the form $(q_0, f(u), A_0[u], q_0) \vdash^*_{\mathcal{A}} (q_0, \epsilon, \epsilon)$. This map $f : X^* \to Y^*$ is a 2-computable sequence.

2.5.2 Catenative recurrent sequences

Definition 21 (catenative recurrent relations) *Given a finite set I and a family of mappings indexed by I, $f_i : X^* \to Y^*$ (for $i \in I$), we call* system of catenative recurrent relations *over the family $(f_i)_{i \in I}$ a system of the form*

$$f_i(xw) = \prod_{j=1}^{\ell(i,x)} f_{\alpha(i,x,j)}(w) \text{ for all } i \in I, x \in X, w \in X^*$$

where $\ell(x, i) \in \mathbb{N}, \alpha(i, x, j) \in I$.

In the particular case where Y is reduced to one letter, a mapping $f : X^* \to Y^*$ is member of a family fulfilling a system of catenative recurrent relations iff f is a $\mathbb{N}$-rational series.

Definition 22 (left-regular catenative recurrent relations) *Let $\equiv$ be an equivalence relation, of finite index, compatible with left product, on X^*. Let $\mathcal{C} = X^*/\equiv$, let I be a finite set and $f_i : X^* \to Y^*$ (for $i \in I$) be a family of mappings. We call* system of catenative recurrent relations *over the family*

$(f_i)_{i\in I}$ *a system of the form*

$$f_i(xw) = \prod_{j=1}^{\ell(i,x,c)} f_{\alpha(i,x,j,c)}(w) \text{ for all } i \in I, x \in X, c \in \mathcal{C}, w \in c$$

where $\ell(x,i,c) \in \mathrm{N}, \alpha(i,x,j,c) \in I$.

2.6 Lindenmayer systems

Let us recall the notion of Lindenmayer system of type HDT0L.

Definition 23 (HDT0L, [KRS97]) *Let* $f : X^* \to Y^*$. *The mapping* f *is called a HDT0L mapping (or sequence) iff there exists a finite alphabet* A, *an homomorphism* $H : X^* \to \mathrm{HOM}(A^*, A^*)$, *an homomorphism* $h \in \mathrm{HOM}(A^*, Y^*)$ *and a letter* $a \in A$ *such that, for every* $w \in X^*$

$$f(w) = h(H^w(a)).$$

(here we denote by H^w the image of w by H).

Proposition 24 *Let* X, Y, Z *be three finite alphabets and* $f : X^* \to Y^*, g : Y^* \to Z^*$ *be some maps. If* f *is a right-sequential map and* g *is a HDT0L, then* $f \circ g$ *is a HDT0L too.*

Proof: According to Definition 23, there exists an intermediate finite alphabet A, an homomorphism $H : Y^* \to \mathrm{HOM}(A^*, A^*)$, an homomorphism $h_0 \in \mathrm{HOM}(A^*, Z^*)$ and a letter $a_0 \in A$ such that, for every $w \in Y^*$

$$g(w) = h_0(H^w(a_0)).$$

The map f is computed by some finite transducer

$$\mathcal{A} := \langle X, Y, Q, Q_-, q_+, \delta \rangle,$$

which is a right-gsm. We use the following general idea: one can simulate the composition $f \circ g$ by a single HDT0L which uses the intermediate alphabet $Q \times A$ (instead of A), some “transition“ homomorphisms K^x (for $x \in X$) that simulate, in parallel, all possible $\mathcal{A}$-transitions $(q, x, v, q') \in \delta$ and the corresponding transition homomorphisms H^v. Let us describe, formally, this new HDT0L.
We define an operation $\otimes : Q \times A^* \to (Q \times A)^*$ by:

$$
\begin{aligned}
q \otimes \varepsilon &:= \varepsilon \\
q \otimes a &:= (q,a) && \text{if } a \in A \\
q \otimes (u \cdot v) &:= (q \otimes u) \cdot (q \otimes v) && \text{if } u, v \in A^*
\end{aligned} \tag{3}
$$

We consider the homomorphism $K : X^* \to Hom((Q \times A)^*, (Q \times A)^*)$ such that, for every $x \in X, q \in Q, a \in A$,

$$K^x(q,a) = \prod_{(q,x,v,q') \in \delta} q' \otimes H^v(a) \tag{4}$$

and, finally, an homomorphism $k_q : (Q \times A)^* \to Z^*$ that selects the letters associated with the state q: for every $q, r \in Q, a \in A$

$$
\begin{aligned}
k_q(r,a) &:= a \quad \text{if } q = r \\
k_q(r,a) &:= \varepsilon \quad \text{if } q \neq r.
\end{aligned}
$$

One can show, by induction over the integer $|u|$ that, for every $u \in X^*, q, r \in Q, v \in A^*$

$$
\begin{aligned}
k_r(K^u(q \otimes v)) &= H^{u'}(v) && \text{if } q \xrightarrow{(u,u')}_{\mathcal{A}} r \\
k_r(K^u(q \otimes v)) &= \varepsilon && \text{if } \forall u', q \not\xrightarrow{(u,u')}_{\mathcal{A}} r.
\end{aligned} \tag{5}
$$

Since $\mathcal{A}$ is co-deterministic, it has a single final state q_+. By the above equations, for every $q \in Q_-$,

$$k_{q_+}(K^u(q \otimes a_0)) = H^{u'}(a_0) \quad \text{if } q \xrightarrow{(u,u')}_{\mathcal{A}} q_+ \tag{6}$$

$$k_{q_+}(K^u(q \otimes a_0)) = \varepsilon \quad \text{if } \forall u', q \not\xrightarrow{(u,u')}_{\mathcal{A}} q_+. \tag{7}$$

Let us define

$$\alpha_0 := \prod_{q \in Q_-} (q, a_0). \tag{8}$$

For every $u \in X^*$, since f is a map, there exists exactly one $q \in Q_-$ such that there exists some word $u' \in Y^*$ such that $q \xrightarrow{(u,u')}_{\mathcal{A}} q_+$. Hence, formulas (7)(8) imply that

$$k_{q_+}(K^u(\alpha_0)) = H^{f(u)}(a_0).$$

Now, the map $u \mapsto k_{q_+}(K^u(\alpha_0))$ fulfills the definition of an HDT0L, excepted for the condition that the initial word α_0 should be a single letter. But it is a routine task to transform such a map, into a true HDT0L. □

Lemma 25 *Let X, Y be two finite alphabets and $f, g : X^* \to Y^*$ be some HDT0L maps. Then $f \cdot g : w \mapsto f(w) \cdot g(w)$ is an HDT0L.*

This result can be easily checked.

3 The main result

The main result of this paper is the following

Theorem 26 *Let X, Y be two finite alphabets, $f : X^* \to Y^*$ a mapping. The following conditions are equivalent :*
1- f belongs to $\mathbb{S}_2(X^*, Y^*)$ i.e. is computed by some strongly deterministic pushdown of pushdown automaton
2- f is a HDT0L
3- f fulfills a system of catenative recurrent equations

This theorem is obtained via the proof of the following more technical result.

Lemma 27 *Let X, Y be two finite alphabets, $f : X^* \to Y^*$ a mapping, then the following conditions are equivalent :*
1- f belongs to $\mathbb{S}_2(X^*, Y^*)$
2- f is a left-regular catenative recurrent sequence
3- f is a right-sequential map composed by a HDT0L
4- f is a HDT0L
5- f is a catenative recurrent sequence.

We prove Lemma 27 in next section. We proceed by proving successively that $(i) \Rightarrow (i + 1)$ for $1 \leq i \leq 4$ and finally, that $(5) \Rightarrow (1)$.

4 The proof

In the following, f will be a map from X^* to Y^*.

4.1 2-computable $\Longrightarrow$ left-regular catenative

Let us suppose that $f : X^* \to Y^*$ is computed by a 2-sdpda $\mathcal{A} = (Q, \Gamma, Y, \delta, q_0, A_0)$ (where $X \subseteq \Gamma$). Without loss of generality, we can suppose that the *push* operations are normalized in such a way that every push move of $\mathcal{A}$ is pushing a word γ of length 2.

Remembering the definition of a left-regular catenative recurrent sequence, we are to choose a finite set I of indices, an equivalence relation, and a system of equations. The choice for I is the following : $I = (Q \times \Gamma \times Q) \cup Y$; the sequence is defined by: $\forall p, q \in Q, A \in \Gamma, y \in Y, u \in X^*$

$$\begin{aligned}
\{S_{(p,A,q)}(u)\} &= \mathrm{L}_{\mathcal{A}}((p, A[u], q)) & \text{if } \mathrm{L}_{\mathcal{A}}((p, A[u], q)) \neq \emptyset \quad (9)\\
S_{(p,A,q)}(u) &= \varepsilon & \text{if } \mathrm{L}_{\mathcal{A}}((p, A[u], q)) = \emptyset \quad (10)\\
S_y(u) &= y & (11)
\end{aligned}$$

Note that, since $\mathcal{A}$ is strongly deterministic, equation (9) really defines a unique word; equation (10) assigns an arbitrary value to those symbols $S_{(p,A,q)}(u)$ corresponding to non-productive words. Thus, every S_i, for $i \in I$ is a map from X^* to Y^*.

The following recurrent assertions hold: $\forall p, q \in Q, A \in \Gamma, x \in X, u \in X^*, \bar{y} \in Y \cup \{\varepsilon\}$,

$$\begin{aligned}
S_{p,A,q}(x \cdot u) &= \bar{y} & \text{if } \delta(p, Ax, \bar{y}) = (q, \mathrm{pop}_1)\\
S_{p,A,q}(x \cdot u) &= \varepsilon & \text{if } \delta(p, Ax, \bar{y}) = (r, \mathrm{pop}_1) \text{ and } q \neq r\\
S_{p,A,q}(x \cdot u) &= \bar{y} \cdot S_{r,A,q}(u) & \text{if } \delta(p, Ax, \bar{y}) = (r, \mathrm{pop}_2)\\
S_{p,A,q}(x \cdot u) &= \bar{y} \cdot S_{r,B,s}(xu) \cdot S_{s,C,q}(xu) & \text{if } \delta(p, Ax, \bar{y}) = (r, \mathrm{push}_1(BC)),\\
& & \text{for some } s \in Q \quad (12)\\
S_{p,A,q}(x \cdot u) &= \bar{y} \cdot S_{r,A,q}(abu) & \text{if } \delta(p, Ax, \bar{y}) = (r, \mathrm{push}_2(ab)). \quad (13)\\
S_y(xu) &= S_y(u) & .
\end{aligned}$$

Nevertheless the two last assertions (12-13) do not meet the general form prescribed by Definition 21 of a catenative recurrence: (12) is not even an equation because the state s is not yet known to be completely determined by the left-hand side;(13) is an equation but the argument in the right-hand side is abu, while it ought to be u.

Our efforts now consist in showing that:
- under the restriction that $\mathrm{L}_{\mathcal{A}}(p, A[x \cdot u], q) \neq \emptyset$, the state s in equation (12) is unique and depends (only) on the class of u for some left-regular equivalence with finite index
- the "increasing" argument abu in equation (13) can be replaced by a finite product of $S_i(u)$ (for some indices $i \in I$), by considering a finite derivation modulo $\rightarrow_{\mathcal{A}}$ starting from $(p, A[xu], q)$ and ending in a finite product of letters from Y or having the form $(p_j, A_j[u], q_j)$.

4.1.0.1 An equivalence of finite index Let us define an equivalence relation over X^*.

Definition 28 *For every $u, v \in X^*$, $u \equiv v$ if and only if*

$$\forall A \in \Gamma, p, q \in Q, \quad \mathrm{L}_{\mathcal{A}}(p, A[u], q) \neq \emptyset \Leftrightarrow \mathrm{L}_{A}(p, A[v], q) \neq \emptyset$$

Let us note $\mathcal{C} := X^*/\equiv$.

Lemma 29 *The equivalence relation $\equiv$ has finite index.*

Proof: It suffices to remark that the class of a word u is characterized by the mapping $(p, A, q) \mapsto 1$ (resp. 0) if $\mathrm{L}_A(p, A[u], q) \neq \emptyset$ (resp. $\mathrm{L}_A(p, A[u], q) = \emptyset$). Hence

$$\mathrm{Card}(\mathcal{C}) \leq 2^{\mathrm{Card}(Q)^2 \times \mathrm{Card}(\Gamma)}.$$

□

Let us introduce some more notation :

- we fix a system $(u_c)_{c \in \mathcal{C}}$ of representatives for the equivalence $\equiv$ (thus every u_c is a word over X which belongs to c)
- $\mathbb{O} = \{\Omega_c \mid c \in \mathcal{C}\}$ is a set of undeterminates, in a one-to-one correspondance with $\mathcal{C}$
- $\pi : (\Gamma \cup \mathbb{O})^* \to (X \cup \Gamma)^*$ is the homomorphism defined by $\pi(\Omega_c) := u_c$ (for every $c \in \mathcal{C}$) and $\pi(\gamma) = \gamma$ (for every $\gamma \in \Gamma$); we call it the projection of the unknowns on X^*
- $E := Q \times 2 - \mathrm{pds}_{\mathbb{O}}(\Gamma) \times Q$

The homomorphism π is transferred as an homomorphism $\pi : E^* \to (Q \times 2 - pds(\Gamma) \times Q)^*$ by:

$$\pi((p, A_1[\omega_1] \cdots A_\ell[\omega_\ell], q)) := (p, A_1[\pi(\omega_1)] \cdots A_\ell[\pi(\omega_\ell)], q).$$

Let $\mathcal{T}$ be the following infinite alphabet :

$$\mathcal{T} = \{e \in E \mid \mathrm{L}_{\mathcal{A}}(\pi(e)) \neq \emptyset\}$$

Lemma 30 (Uniqueness of the second state) *Let $q \in Q, \omega \in 2{-}\mathrm{pds}_{\mathbb{O}}(\Gamma)$ and $r, r' \in Q$ such that $(q, \omega, r), (q, \omega, r') \in \mathcal{T}$. Then $r = r'$.*

Proof: Suppose that

$$(q, \pi(\omega), r) \to_{\mathcal{A}}^* u \in Y^*, \;\; (q, \pi(\omega), r') \to_{\mathcal{A}}^* u' \in Y^*.$$

By Lemma 16 we have

$$(q, u, \pi(\omega)) \vdash_{\mathcal{A}}^* (r, \epsilon, \epsilon), \;\; (q, u', \pi(\omega)) \vdash_{\mathcal{A}}^* (r', \epsilon, \epsilon).$$

The automaton $\mathcal{A}$ is *strongly* deterministic. This implies that there is at most one computation of $\mathcal{A}$ starting on a configuration of the form $(q, *, \pi(\omega))$ (for a given ω) and ending on a configuration of the form $(*, \varepsilon, \varepsilon)$. Hence $u = u'$ and $r = r'$. $\square$

Lemma 31 (The decomposition state) *Let* $(p, \omega_1 \cdot \omega_2, q) \in \mathcal{T}$ *with* $\omega_1 \neq \epsilon \wedge \omega_2 \neq \epsilon$. *Then*

$$\exists! r \in Q, (p, \omega_1, r) \in \mathcal{T} \text{ and } (r, \omega_2, q) \in \mathcal{T}$$

Proof: Let $\omega, \omega_1, \omega_2 \in 2 - pds_{\mathbb{O}}(\Gamma), p, q \in Q$ such that $\omega_1 \neq \epsilon, \omega_2 \neq \epsilon$ and $(p, \omega_1 \cdot \omega_2, q) \in \mathcal{T}$.
Existence:
There exists some word $u \in Y^*$ such that

$$(p, \pi(\omega_1) \cdot \pi(\omega_2), q) \to_{\mathcal{A}}^* u,$$

i.e., using Lemma 16 such that

$$(p, u, \pi(\omega_1) \cdot \pi(\omega_2)) \vdash_{\mathcal{A}}^* (q, \varepsilon, \varepsilon).$$

Since $\mathcal{A}$ is a pushdown automaton (over the pushdown alphabet $\{A[\alpha] \mid A \in \Gamma, \alpha \in \Gamma^*\}$), the above computation admits a decomposition of the form

$$(p, u_1, \pi(\omega_1)) \vdash_{\mathcal{A}}^* (r, \varepsilon, \varepsilon), \;\; (r, u_2, \pi(\omega_2)) \vdash_{\mathcal{A}}^* (q, \varepsilon, \varepsilon)$$

for some $q \in Q, u_1, u_2 \in Y^*$. Using again Lemma 16 we conclude that

$$(p, \omega_1, r) \in \mathcal{T} \text{ and } (r, \omega_2, q) \in \mathcal{T}.$$

Unicity:
Suppose that $r' \in Q$ and

$$(p, \omega_1, r') \in \mathcal{T} \text{ and } (r', \omega_2, q) \in \mathcal{T}.$$

Hence we get:

$$(p, \omega_1, r) \in \mathcal{T} \text{ and } (p, \omega_1, r') \in \mathcal{T},$$

which, by Lemma 30 implies that

$$r = r'.$$

$\square$

4.1.0.2 Derivation within $(\mathcal{T} \cup Y)^*$

Definition 32 (restricted derivation) *We define the binary relation $\rightsquigarrow_{\mathcal{A}}$ over $(\mathcal{T} \cup Y)^*$ as the one-step rewriting relation associated to the following semi-Thue system:*

$$\begin{aligned}
(p, \omega, q) &\rightsquigarrow_{\mathcal{A}} \quad \bar{y}(p', \omega', q) && \textit{if } (p, \bar{y}, \omega) \vdash_{\mathcal{A}} (p', \epsilon, \omega') && (14)\\
(p, \omega, q) &\rightsquigarrow_{\mathcal{A}} (p, \omega_1, r)(r, \omega_2, q) && \textit{if } \omega_1 \cdot \omega_2 = \omega, \omega_1 \neq \varepsilon, \omega_2 \neq \varepsilon, &&\\
& && (p, \omega_1, r), (r, \omega_2, q) \in \mathcal{T} && (15)\\
(p, \varepsilon, p) &\rightsquigarrow_{\mathcal{A}} \qquad \varepsilon && \textit{if } p \in Q. && (16)
\end{aligned}$$

Note that:
- every right-hand side of a rule of type (14) belongs to $(\mathcal{T} \cup Y)^*$: since $(p, \omega, q) \in \mathcal{T}$ and $\mathcal{A}$ is strongly deterministic, every computation starting on $(p, \bar{y}*, \omega)$ must go through the configuration $(p', *, \omega')$
- Lemma 31 ensures that, for every $(p, \omega, q) \in \mathcal{T}$ and every decomposition $\omega = \omega_1 \cdot \omega_2$ into non-empty factors, $\rightsquigarrow_{\mathcal{A}}$ admits one rule with left-hand side (p, ω, q) of type (15).

Fact 33 $\rightsquigarrow_{\mathcal{A}}^* \subseteq \rightarrow_{\mathcal{A}}^*$

Proof: The semi-Thue system defining $\rightsquigarrow_{\mathcal{A}}$ consists of exactly those rules from the semi-Thue system defining $\rightarrow_{\mathcal{A}}$ (see Definition 15) where the two hand-sides belong to $(\mathcal{T} \cup Y)^*$. □

Lemma 34 *The relation $\rightsquigarrow_{\mathcal{A}}$ is noetherian*

Proof: Suppose that there exists some $w \in (\mathcal{T} \cup Y)^*$ such that:

$$w \rightsquigarrow_{\mathcal{A}}^{\infty} . \qquad (17)$$

Since the semi-Thue system defining $\rightsquigarrow_{\mathcal{A}}$ is monadic (i.e. its left-hand sides have length ≤ 1), every derivation starting from w is a parallel composition of derivations starting from the letters $w_1, w_2, \ldots, w_\ell \in (\mathcal{T} \cup Y)$ composing the word w. Since there are only finitely many such derivations, one of them must be infinite: $\exists U \in (\mathcal{T} \cup Y)$ such that

$$U \rightsquigarrow_{\mathcal{A}}^{\infty} .$$

But $U \in Y$ is impossible, hence there exist $p, q \in Q, \omega \in 2 - \mathrm{pds}_{\mathbb{O}}(\Gamma)$ such that

$$(p, \omega, q) \rightsquigarrow_{\mathcal{A}}^{\infty} .$$

By Lemma 18, it follows that

$$(p, \pi(\omega), q) \rightsquigarrow_{\mathcal{A}}^{\infty} . \qquad (18)$$

Case 1: Derivation (18) uses infinitely many rules of type (14) (i.e. transition rules).
In particular,

$$\forall n \in \mathrm{N}, \exists u \in Y^*, (p, \pi(\omega), q)(\to_{\mathcal{A},1} \cup \to_{\mathcal{A}}^*)^{\geq n} u$$

because $(p, \pi(\omega), q)(\to_{\mathcal{A},1} \cup \to_{\mathcal{A}}^*)^n u' \in (\mathcal{T} \cup Y)^*$ and every letter of u' that belongs to $\mathcal{T}$ can be derived into a word from Y^*. Hence, by Lemma 17,

$$\forall n \in \mathrm{N}, \exists u \in Y^*, (p, u, \pi(\omega)) \vdash_{\mathcal{A}}^{\geq n} (q, \varepsilon, \varepsilon)$$

It follows that there exist integers $n < m$, words $u_n, u_m \in Y^*$ and states $q_n, q_m \in Q$ such that

$$(p, u_n, \pi(\omega)) \vdash_{\mathcal{A}}^{n} (q_n, \varepsilon, \varepsilon), \quad (p, u_m, \pi(\omega)) \vdash_{\mathcal{A}}^{m} (q_m, \varepsilon, \varepsilon)$$

but, since $\mathcal{A}$ is strongly deterministic, two such different computations are impossible.
Case 2: Derivation (18) uses finitely many rules of type (14).
Let us consider the derivation (18):

$$(p, \pi(\omega), q) = U_0 \rightsquigarrow_{\mathcal{A}} U_1 \ldots \rightsquigarrow_{\mathcal{A}} U_n \rightsquigarrow_{\mathcal{A}} U_{n+1} \ldots$$

For every $U \in (\mathcal{T} \cup Y)^*$ we define

$$H(U) = H_\varepsilon(U) - \bar{H}_\varepsilon(U)$$

where $H_\varepsilon(U)$ is the number of ocurrences in U of letters from the subalphabet $\{(p, \varepsilon, p) \mid p \in Q\}$ and $\bar{H}_\varepsilon(U)$ is the number of ocurrences in U of letters from the subalphabet $\{(p, \omega, q) \mid p, q \in Q, \omega \in 2-\mathrm{pds}(\Gamma), \omega \neq \varepsilon\}$. Every rule of type (15) decreases strictly (resp. preserves) the number $(-\bar{H}_\varepsilon(U))$ (resp. $H_\varepsilon(U)$) while every rule of type (16) decreases strictly the number $H_\varepsilon(U)$ (resp. preserves $(-\bar{H}_\varepsilon(U))$). Hence, every rule of type (15) or (16) strictly decreases the number $H(U)$.
There exists some n_0 such that, only rules of type (15)(16) are used in the infinite derivation

$$U_{n_0} \rightsquigarrow_{\mathcal{A}} U_{n_0+1} \ldots \rightsquigarrow_{\mathcal{A}} U_n \rightsquigarrow_{\mathcal{A}} U_{n+1} \ldots$$

and the numbers $H(U_n)$, for $n \geq n_0$ are strictly decreasing. Let us consider the decomposition of U_{n_0} into letters of $\mathcal{T} \cup Y$:

$$U_{n_0} = v_0 \prod_{i=1}^{\ell_0} (p_i, \omega_i, q_i) v_i$$

where $(p_i, \omega_i, q_i) \in \mathcal{T}$ and $v_i \in Y^*$. Let us consider the natural integer

$$N := \sum_{i=1}^{\ell_0} |\omega_i|.$$

Clearly, $\forall n \geq n_0, H_\varepsilon(U_n) \geq 0$. Moreover, since the rules (15)(16) can only either decompose or remove some letters from $\mathcal{T}$, $\forall n \geq n_0, \bar{H}_\varepsilon(U_n) \leq N$. It follows that

$$\forall n \geq n_0, H(U_n) \geq -N,$$

which contradicts the fact the sequence of integers $(H(U_n))_{n \geq n_0}$ is strictly decreasing. Finally, the existence of an infinite sequence (17) is impossible. □

Lemma 35 *The relation $\leadsto_{\mathcal{A}}$ is strongly locally confluent.*

Proof: It suffices to check that every "criticial pair" of the semi-Thue system defining $\leadsto_{\mathcal{A}}$ is resolved by derivations of length ≤ 1. Since all the left-hand sides of rules (14–16) are letters, the only critical pairs that arise have the following form:

$$(p, \omega_1\omega_2, q) \leadsto_{\mathcal{A}} \bar{y}(p', \mathrm{op}(\omega_1\omega_2), q) \tag{19}$$
$$(p, \omega_1\omega_2, q) \leadsto_{\mathcal{A}} (p, \omega_1, r)(r, \omega_2, q) \tag{20}$$

where $\delta(p, \mathit{topsymb}(\omega), \bar{y}) = (p', \mathrm{op}), r \in Q, (p, \omega_1, r), (r, \omega_2, q) \in \mathcal{T}$.
Note that the above derivations warranty that $(p', \mathrm{op}(\omega_1\omega_2), q), (p, \omega_1, r), (r, \omega_2, q) \in \mathcal{T}$. Taking into account the fact that $\omega_1 \neq \varepsilon$, we also have:

$$\mathrm{op}(\omega_1\omega_2) = \mathrm{op}(\omega_1)\omega_2$$

(whatever the precise operation $\mathrm{op} \in \mathrm{POP} \cup \mathrm{PUSH}(\Gamma)$ is),

$$(p', \mathrm{op}(\omega_1)\omega_2), q) \leadsto_{\mathcal{A}} (p', \mathrm{op}(\omega_1), r)(r, \omega_2, q)$$

(because the two letters in rhs belong to $\mathcal{T}$), and

$$(p, \omega_1, r) \leadsto_{\mathcal{A}} \bar{y}(p', \mathrm{op}(\omega_1), r).$$

Hence

$$\begin{array}{rcl} \bar{y}(p', \mathrm{op}(\omega_1\omega_2), q) & = & \bar{y}(p', \mathrm{op}(\omega_1)\omega_2), q) \leadsto_{\mathcal{A}} \bar{y}(p', \mathrm{op}(\omega_1), r)(r, \omega_2, q) \\ & & \text{by a decomposition rule} \\ (p, \omega_1, r)(r, \omega_2, q) & \leadsto_{\mathcal{A}} & \bar{y}(p', \mathrm{op}(\omega_1), r)(r, \omega_2, q) \\ & & \text{by a transition rule} \end{array}$$

which is a resolution of the critical pair (19)(20) by derivations of lengths ≤ 1.
□

Lemma 36 *The relation $\leadsto_{\mathcal{A}}$ is confluent and noetherian.*

Proof: By Lemma 35 and Lemma 1, $\rightsquigarrow_{\mathcal{A}}$ is confluent and by Lemma 34, $\rightsquigarrow_{\mathcal{A}}$ is noetherian. $\square$

Lemma 37 $\equiv$ *is a left-regular equivalence relation.*

Proof: Let $v, v', w \in X^*$, such that :

$$v \equiv v'.$$

Let $c := [v]_\equiv = [v']_\equiv$. Let $p, q \in Q$ and $A \in \Gamma$ such that :

$$L_{\mathcal{A}}(p, A[w \cdot v], q) \neq \emptyset \tag{21}$$

The letter $(p, A[w \cdot \Omega_c], q)$ belongs to $\mathcal{T}$, since its image by π generates a word in Y^* (by (21) and the assumption that $u_c \equiv v$). Let us consider the normal form of $(p, A[w \cdot \Omega_c], q)$ for the relation $\rightsquigarrow_{\mathcal{A}}$:

$$(p, A[w \cdot \Omega_c], q) \rightsquigarrow_{\mathcal{A}}^* \rho_{\rightsquigarrow_{\mathcal{A}}}((p, A[w \cdot \Omega_c], q)) = \prod_{i=0}^{m} W_i \tag{22}$$

where, for every $i \in [0, m]$, $W_i \in (Y \cup \mathcal{T})$. But the form of the rules of $\rightsquigarrow$ implies that those W_i which belong to $\mathcal{T}$ have the form

$$W_i = (p_i, A_i[\omega_i \Omega_c], q_i)$$

where $\omega_i \in \Gamma^*$. The fact that $(p_i, A_i[\omega_i \Omega_c], q_i) \in \mathcal{T}$ and is an irreducible letter (w.r.t. $\rightsquigarrow_{\mathcal{A}}$), implies that $\omega_i = \varepsilon$. Finally, rewriting separately the letters from Y, the derivation (22) can be described as

$$(p, A[w \cdot \Omega_c], q) \rightsquigarrow_{\mathcal{A}}^* v_0 \prod_{i=1}^{n} (p_i, A_i[\Omega_c], q_i) v_i \tag{23}$$

where $v_i \in Y^*$ (for $0 \leq i \leq n$), $(p_i, A_i[\Omega_c], q_i) \in \mathcal{T}$ (for $1 \leq i \leq n$). By the substitution principle (Lemma 18), derivation (23) implies that

$$(p, A[w \cdot v'], q) \rightsquigarrow_{\mathcal{A}}^* v_0 \prod_{i=1}^{n} (p_i, A_i[v'], q_i) v_i \tag{24}$$

Since $(p_i, A_i[\Omega_c], q_i) \in \mathcal{T}$, by definition, $\mathrm{L}_{\mathcal{A}}(p_i, A_i[u_c], q_i) \neq \emptyset$ and, since $u_c \equiv v'$,

$$\mathrm{L}_{\mathcal{A}}(p_i, A_i[v'], q_i) \neq \emptyset \tag{25}$$

Derivation (24) together with (25) implies that

$$L_{\mathcal{A}}(p, A[w \cdot v'], q) \neq \emptyset \tag{26}$$

Of course , by a similar argument, (26) implies , conversely, (21), hence

$$w \cdot v \equiv w \cdot v'.$$

□

4.1.0.3 The system of catenative recurrent equations

Let us consider the set of indices $I = (Q \times \Gamma \times Q) \cup Y$ and the family of maps $(S_i)_{i \in I}$ defined at the beginning of this subsection (see equalities (9–11)). Let us use the left-regular equivalence $\equiv$ and the key-idea of the proof of Lemma 37. For every $p, q \in Q, A \in \Gamma, c \in \mathcal{C}, x \in X$ such that $(p, A[x\Omega_c], q) \in \mathcal{T}$, the normal form $\rho_{\rightsquigarrow_{\mathcal{A}}}(p, A[x\Omega_c], q)$ must have a decomposition of the form

$$\rho_{\rightsquigarrow_{\mathcal{A}}}(p, A[x\Omega_c], q) = \prod_{j=1}^{\ell((p,A,q),x,c)} W_{(p,A,q),x,j,c} \tag{27}$$

where $W_{(p,A,q),x,j,c} \in Y \cup \mathcal{T}$. The index $\alpha((p, A, q), x, j, c)$ is then defined by

$$\alpha((p, A, q), x, j, c) := (p', A', q') \text{ if } W_{(p,A,q),x,j,c} = (p', A'[\Omega_c], q') \in \mathcal{T} \tag{28}$$

$$\alpha((p, A, q), x, j, c) := \quad y \quad \text{ if } W_{(p,A,q),x,j,c} = y \in Y \tag{29}$$

Let us consider now the following system of catenative recurrent equations: $\forall p, q \in Q, A \in \Gamma, x \in X, c \in C, u \in c$

$$S_{p,A,q}(x \cdot u) = \prod_{j=1}^{\ell((p,A,q),x,c)} S_{\alpha((p,A,q),x,j,c)}(u) \tag{30}$$

if $(p, A[x\Omega_c], q) \in \mathcal{T}$,

$$S_{p,A,q}(x \cdot u) = \varepsilon = \prod_{j=1}^{0} S_{\alpha((p,A,q),x,j,c)}(u) \tag{31}$$

if $(p, A[x\Omega_c], q) \notin \mathcal{T}$,

$$S_y(x \cdot u) = S_y(u) \tag{32}$$

if $y \in Y$.
This sytem has the form required by Definition (21), if we define $\ell((p, A, q), x, c) := 0$ when $(p, A[x\Omega_c], q) \notin \mathcal{T}$ and $\ell(y, x, c) := 1$ when $y \in Y$. Let us check that the family $(S_i)_{i \in I}$ does fulfill equations (30–32):
- equality (27) shows that, when $(p, A[x\Omega_c], q) \in \mathcal{T}$,

$$(p, A[x\Omega_c], q) \rightarrow^*_A \prod_{j=1}^{\ell((p,A,q),x,c)} W_{(p,A,q),x,j,c}$$

which, by the substitution principle entails that

$$(p, A[xu], q) \to_A^* \prod_{j=1}^{\ell((p,A,q),x,c)} W_{(p,A,q),x,j,c}[u/\Omega_c]$$

hence equality (30)
- equality (31) is due to the fact that, when $(p, A[x\Omega_c], q) \notin \mathcal{T}$, $S_{p,A,q}(x \cdot u) = \varepsilon$
- equality (32) is obviously true (the sequences S_y are constant). (End of the proof that (1) $\Rightarrow$ (2) in Lemma 27).

4.2 Left-regular catenative $\Longrightarrow$ gsm composed by a HDT0L

Suppose that $f = f_{i_0}$ where $(f_i)_{i \in I}$ is a family of maps $X^* \to Y^*$ which are solution of a system of left-regular catenative recurrent equations of the form (22). We recall such a system uses some left-regular equivalence $\sim$, over X^*, which has finite index. We note $(u, c) \mapsto u \odot c$ the left-action of X^* over $X^*/\sim$ defined by $(u, [v]_\sim) \mapsto [uv]_\sim$.
Let us consider the intermediate alphabet $Z := X \times \mathcal{C}$ and recall that the map $\mathcal{G}_\sim : X^* \to Z^*$ introduced by Definition 4 is right-sequential. Let us build an HDT0L $g : Z^* \to Y^*$ such that

$$f = \mathcal{G}_\sim \circ g.$$

Let $\pi_i : Z^* \to X^*$ be the i-th projection (for $1 \le i \le 2$), i.e.

$$\forall x \in X, c \in \mathcal{C}, \;\; \pi_1(x, c) := x, \;\; \pi_2(x, c) := c.$$

Let us consider the new set of indexes $K := I \times \mathcal{C}$ and the family of sequences, indexed over K, $g_{i,c} : Z^* \to Y^*$ defined by

$$g_{i,c}(w) := f_i(\pi_1(w)) \text{ if } w \in \mathrm{im}(\mathcal{G}_\sim) \text{ and } [\pi_1(w)]_\sim = c \tag{33}$$
$$g_{i,c}(w) := \quad \varepsilon \quad \text{ if } w \notin \mathrm{im}(\mathcal{G}_\sim) \text{ or } [\pi_1(w)]_\sim \neq c. \tag{34}$$

Let us consider the system of recurrence relations

$$g_{i,d}((x, c)w) = \prod_{j=1}^{\ell(i,x,c)} g_{\alpha(i,x,j,c),c}(w) \text{ for all } i \in I, d \in \mathcal{C}, x \in X, c \in \mathcal{C}, w \in Z^* \text{ such that } d = x \odot c \tag{35}$$
$$g_{i,d}((x, c)w) = \varepsilon \quad \text{for all } i \in I, d \in \mathcal{C}, x \in X, c \in \mathcal{C}, w \in Z^* \text{ such that } d \neq x \odot c \tag{36}$$

Lemma 38 *The family of maps $(g_k)_{k \in K}$ fulfills the* system of recurrent relations *(35–36).*

Proof: Let $i \in I, d \in \mathcal{C}, x \in X, c \in \mathcal{C}, w \in Z^*$. We distinguish several cases and check that (35–36) holds in every case.
Case 1:$w \notin \mathrm{im}(\mathcal{G}_\sim)$.
Under these hypotheses, $(x,c)w \notin \mathrm{im}(\mathcal{G}_\sim)$, hence equality (34) applies i.e. $g_{i,d}((x,c)w) = \varepsilon$.
The rhs of relation (35) is a product of sequences the value of which is ε, hence its value is ε while rhs of relation (36) is explicitly ε.
Case 2:$c \neq [\pi_1(w)]_\sim$, $w \in \mathrm{im}(\mathcal{G}_\sim)$.
In this case also $(x,c)w \notin \mathrm{im}(\mathcal{G}_\sim)$, so that $g_{i,d}((x,c)w) = \varepsilon$. All the $g_{\alpha(i,x,j,c),c}(w)$ are equal to ε (by(34)). We can conclude as in Case 1.
Case 3:$d \neq x \odot c$, $c = [\pi_1(w)]_\sim$, $w \in \mathrm{im}(\mathcal{G}_\sim)$.
Since $d \neq x \odot c = [\pi_1((x,c)w)]_\sim$, by Definition (34), $g_{i,d}((x,c)w) = \varepsilon$, hence relation (36) is fulfilled.
Case 4:$d = x \odot c$, $c = [\pi_1(w)]_\sim$, $w \in \mathrm{im}(\mathcal{G}_\sim)$.
In this case

$$g_{i,d}((x,c)w) = f_i(x\pi_1(w))$$

and for every $j \in [1, \ell(i,x,c)]$,

$$g_{\alpha(i,x,j,c),c}(w) = f_{\alpha(i,x,j,c)}(\pi_1(w))$$

and since

$$f_i(x\pi_1(w)) = \prod_{j=1}^{\ell(i,x,c)} f_{\alpha(i,x,j,c)}(\pi_1(w))$$

relation (35) holds. □

Lemma 39 *The* system of recurrent relations *(35–36) is a system of catenative recurrent relations.*

Proof: Let us define

$$\begin{array}{rcl} \ell'((i,d),(x,c) := & \ell(i,x,c) & \text{if } d = x \odot c \\ \ell'((i,d),(x,c) := & 0 & \text{if } d \neq x \odot c \\ \alpha'((i,d),(x,c),j) := & (\alpha(i,x,j,c),c) & \text{if } d = x \odot c \end{array}$$

The system (35–36) can thus be rewritten under the form

$$g_{i,d}((x,c)w) = \prod_{j=1}^{\ell'((i,d),(x,c))} g_{\alpha'((i,d),(x,c),j)}(w) \text{ for all } i \in I, d \in \mathcal{C}, x \in X, c \in \mathcal{C}, w \in Z^*$$

which is a *system of catenative recurrent relations* in the sense of Definition 21. □

Lemma 40 *Every catenative recurrent sequence is a HDT0L.*

Proof: Let $(f_i)_{i\in I}$ be a family of maps $f_i : X^* \to Y^*$, fulfilling a system of the form

$$f_i(xw) = \prod_{j=1}^{\ell(i,x)} f_{\alpha(i,x,j)}(w) \text{ for all } i \in I, x \in X, w \in X^*$$

We define a finite alphabet A and homomorphisms $H : X^* \to \mathrm{HOM}(A^*, A^*), h : A^* \to Y^*$ by:

$$A := \{a_i \mid i \in I\}$$

i.e. A is an alphabet, in bijection with I; for every $x \in X$,

$$H^x : a_i \mapsto \prod_{j=1}^{\ell(i,x)} a_{\alpha(i,x,j)}$$

and

$$h : a_i \mapsto f_i(\varepsilon).$$

Let us check that

$$\forall i \in I, \forall w \in X^*, h(H^w(a_i)) = f_i(w). \tag{37}$$

For $w = \varepsilon$ we have:

$$h(H^\varepsilon(a_i)) = h(a_i) = f_i(\varepsilon).$$

For every $w \in X^*, x \in X$

$$\begin{aligned} h(H^{xw}(a_i)) = \quad & h(H^w(H^x(a_i))) \\ = & h(H^w(\prod_{j=1}^{\ell(i,x)} a_{\alpha(i,x,j)})) \\ = & \prod_{j=1}^{\ell(i,x)} h(H^w(a_{\alpha(i,x,j)})) \end{aligned} \tag{38}$$

Hence both families of sequences $(w \mapsto h(H^w(a_i)))_{i\in I}$, $(f_i)_{i\in I}$ satisfy the same catenative recurrence relations and start with the same value on ε. Hence they are equal, i.e. (37) holds. By definition, the map $w \mapsto h(H^w(a_i))$ is a HDT0L. $\square$

By Lemma 38 combined with Lemma 40 $(g_k)_{k\in K}$ are all HDT0L maps.

Note that, on a given word $w \in \mathrm{im}(\mathcal{G}_\sim)$, and for a given index $i \in I$, the disjunction of cases used in (33–34) results in the fact that there is exactly one class $c \in \mathcal{C}$ such that $f_i(w) = g_{i,c}(\mathcal{G}_\sim(w))$, while all the others $g_{i,c}(\mathcal{G}_\sim(w))$ evaluate to ε. Hence

$$f_i(w) = \prod_{c\in\mathcal{C}} g_{i,c}(\mathcal{G}_\sim(w)) \tag{39}$$

Let us define

$$g := \prod_{c\in\mathcal{C}} g_{i,c}$$

By Lemma 25, g is an HDT0L. By equality (39), $f_{i_0} = \mathcal{G}_\sim \circ g$, hence f_{i_0} is the composition of a right-sequential map by a HDT0L. (End of the proof that (2) $\Rightarrow$ (3) in Lemma 27).

4.3 gsm composed by a HDT0L $\Longrightarrow$ HDT0L

We have to prove that (3) $\Rightarrow$ (4) (in Lemma 27). This follows from Proposition 24.

4.4 HDT0L $\Longrightarrow$ catenative

We have to prove that (4) $\Rightarrow$ (5) (in Lemma 27).
Let us assume a HDTOL map f defined through a finite alphabet $A := \{a_i \mid i \in I\}$, (where I is a finite index set) and homomorphisms $H : X^* \to \mathrm{HOM}(A^*, A^*), h : A^* \to Y^*$ by:

$$f : w \mapsto h(H^w(a_{i_0}))$$

for some $i_0 \in I$. We define more generally, for every $i \in I$, an HDT0L f_i by:

$$f_i : w \mapsto h(H^w(a_i)).$$

Let us define the numbers $\ell(i, x) \in \mathrm{N}$ and choose the indexes $\alpha(i, x, j) \in I$ for $1 \leq j \leq \ell(i, x)$, in such a way that

$$\ell(i, x) :=\mid H^x(a_i) \mid, \quad H^x(a_i) = \prod_{j=1}^{\ell(i,x)} a_{\alpha(i,x,j)}.$$

The calculation (38) shows that the maps f_i fulfill the relations

$$f_i(xw) = \prod_{j=1}^{\ell(i,x)} f_{\alpha(i,x,j)}(w)$$

which constitute a system of catenative recurrent equations.
(End of the proof that (4) $\Rightarrow$ (5) in Lemma 27).

4.5 catenative $\Longrightarrow$ 2-computable

We have to prove that (5) $\Rightarrow$ (1) (in Lemma 27).
Let f be a catenative recurrent map. This means $f = f_{i_0}$ where $(f_i)_{i \in I}$ is a family, indexed over a finite set I, of mappings $f_i : X^* \to Y^*$ fulfilling a system

of recurrence relations of the form (21). Let us denote by $y_{i,j} \in Y$ the j-th letter of the word $f_i(\varepsilon)$ i.e.

$$f_i(\varepsilon) = \prod_{j=1}^{|f_i(\varepsilon)|} y_{i,j}.$$

We define a pushdown automaton of level 2 $\mathcal{A} = (Q, Y, \Gamma, \delta, q_0, A_{i_0})$ by :

$$Q := \{q_0\} \cup \{q_{i,x} \mid i \in I, x \in X\} \cup \{r_{i,j} \mid i \in I, 0 \leq j \leq |f_i(\varepsilon)|\},$$

$$\Gamma := X \cup \{A_i \mid i \in I\},$$

$$\begin{array}{rcll}
\delta(q_0, \varepsilon, A_i x) := & (q_{i,x}, \mathrm{pop}_2) & \text{for } i \in I, x \in X & (40) \\
\delta(q_{i,x}, \varepsilon, A_i \bar{a}) := & (q_0, \mathrm{push}_1(\prod_{j=1}^{\ell(i,x)} A_{\alpha(i,x,j)})) & \text{for } i \in I, \bar{a} \in X \cup \{\varepsilon\} & (41) \\
\delta(q_0, \varepsilon, A_i) := & (r_{i,0}, \mathrm{push}_1(A_i) & \text{for } i \in I, & (42) \\
\delta(r_{i,j}, y_{i,j+1}, A_i) := & (r_{i,j+1}, A_i) & \text{for } i \in I, 0 \leq j < |f_i(\varepsilon)| & (43) \\
\delta(r_{i,j}, \varepsilon, A_i) := & (q_0, \mathrm{pop}_1) & \text{for } i \in I, j = |f_i(\varepsilon)| & (44)
\end{array}$$

One can easily check that this automaton is strongly deterministic.

Lemma 41 *For every $w \in X^*$ and every $i \in I$, $(q_0, A_i[w], q_0) \to_{\mathcal{A}}^* f_i(w)$.*

Proof: We prove this property by induction over $|w|$.
Base: $w = \varepsilon$.

$$\begin{array}{rcll}
(q_0, A_i[\varepsilon], q_0) \to_{\mathcal{A}} & (r_{i,0}, A_i[\varepsilon], q_0) & \text{by a transition of type (42)} \\
\to_{\mathcal{A}}^* & f_i(\varepsilon)(q_{i,|f_i(\varepsilon)|}, A_i[\varepsilon], q_0) & \text{by transitions of type (43)} \\
\to_{\mathcal{A}} & f_i(\varepsilon)(q_0, \varepsilon, q_0) & \text{by a transition of type (44)} \\
\to_{\mathcal{A}} & f_i(\varepsilon) & \text{by an epsilon rule}
\end{array}$$

Induction step: $w = xv$ for some $x \in X, v \in X^*$.

$$\begin{array}{rcll}
(q_0, A_i[xv], q_0) \to_{\mathcal{A}} & (q_{i,x}, A_i[v], q_0) & \text{by a transition of type (40)} \\
\to_{\mathcal{A}} & (q_0, \prod_{j=1}^{\ell(i,x)} A_{\alpha(i,x,j)}[v], q_0) & \text{by a transition of type (41)} \\
\to_{\mathcal{A}}^* & \prod_{j=1}^{\ell(i,x)} (q_0, A_{\alpha(i,x,j)}[v], q_0) & \text{by decomposition rules .}
\end{array}$$

while, by inuction hypothesis, for every $j \in [1, \ell(i,x)]$

$$(q_0, A_{\alpha(i,x,j)}[v], q_0) \to_{\mathcal{A}}^{*} f_{\alpha(i,x,j)}(v).$$

It follows that

$$(q_0, A_i[xv], q_0) \to_{\mathcal{A}}^{*} \prod_{j=1}^{\ell(i,x)} f_{\alpha(i,x,j)}(v)$$

where, by the assumed recurrence relations, the rhs is exactly $f_i(xv) = f_i(w)$. $\square$

(End of the proof that (5) $\Rightarrow$ (1) in Lemma 27).

Note that the 2-*spda* exhibited in the above proof of (5) $\Rightarrow$ (1) does not use any operation in the set $\{\mathrm{push}_2(\gamma) \mid \gamma \in \Gamma^*\}$. Thus we also have proved the following *normalisation* lemma.

Lemma 42 (normalization of $2-\mathrm{sdpda}$) *For every $2-$ sdpda $\mathcal{A}$ computing a mapping $f : X^* \to Y^*$, it is possible to transform (effectively) $\mathcal{A}$ into a $2-$ sdpda $\mathcal{A}'$ which computes the same mapping f and does not use any* push_2*-operation.*

Proof: Let $\mathcal{A}$ be some $2-\mathrm{sdpda}$ computing a mapping $f : X^* \to Y^*$. By the proof of Lemma 27, we can construct, from the automaton $\mathcal{A}$ a system of catenative recurrence relations that characterize f. From the above proof of (5) $\Rightarrow$ (1), we can extract from these recurrence relations a $2-\mathrm{sdpda}$ $\mathcal{A}'$ which does not use any push_2-operation. $\square$

Example: Let us illustrate the main theorem by some classical sequences. We consider the following HDT0L:

$X := \{x, y\}, Y := \{a, b\}, \Phi : X^* \to \mathrm{HOM}(Y^*, Y^*)$ is defined by

$$\Phi^x(a) := ab, \;\; \Phi^x(b) := ba, \;\; \Phi^y(a) := b, \;\; \Phi^y(b) := ba.$$

We obtain the family $(f_i)_{i\in\{0,1\}}$ of HDT0L maps:

$$f_0(u) := \Phi^u(a), \;\; f_1(u) := \Phi^u(b).$$

(Note that $n \mapsto f_0(x^n)$ is the Thue-Morse sequence and $n \mapsto f_0(y^n)$ is the Fibonacci sequence).

These maps fulfill the system of catenative relations:

$$\begin{cases} f_0(xu) = f_0(u)f_1(u) \\ f_0(yu) = f_1(u) \\ f_1(xu) = f_1(u)f_0(u) \\ f_1(yu) = f_1(u)f_0(u) \end{cases}$$

These maps are computed by the $2-$sdpda $\mathcal{A}$ given in the example following Definition 15: for every $i \in \{0,1\}$:

$$(q_0, A_i[u], q_0) \to_{\mathcal{A}}^{*} f_i(u).$$

5 Some applications

5.1 The skimming theorem for $\mathbb{N}$-rational series

The following statement is known as the "skimming theorem"

Theorem 43 *Let Y be a finite alphabet, $g : Y^* \to \mathbb{N}$ be some map and $g' : Y^* \to \mathbb{N}$ be defined by:*

$$g'(u) := max\{0, f(u) - 1\}$$

If g is a $\mathbb{N}$-rational series, then g' is $\mathbb{N}$-rational too.

This statement appeared as Theorem 3 in [Sch70] [2]. It is also exposed in [Sak03] and an improved proof together with applications is given in [SdS08]. With the help of Theorem 26, we give yet another proof of the skimming theorem; in a second step we also extend the skimming theorem to a more general statement dealing with HDT0L's (which we see as a natural extension of $\mathbb{N}$-rational series).

Proof: Suppose $g : Y^* \to \mathbb{N}$ is a rational series. The linear recurrent equations (with coefficients in $\mathbb{N}$) that define g can be seen as a system of catenative recurrent equations defining $g : Y^* \to \{z\}^*$ (we identify here every integer n with the word z^n). Hence, by Theorem 26, g is computed by some sdpda $\mathcal{A} = (Q, \{z\}, \Gamma, \delta, q_0, A_0)$: for all $u \in Y^*$:

$$(q_0, g(u), A_0[u]) \vdash_{\mathcal{A}}^{*} (q_0, \varepsilon, \varepsilon).$$

Let us construct a $2-$sdpda $\mathcal{A}'$ which simulates $\mathcal{A}$ but omits the first transition that reads a letter on the tape.

The sdpda $\mathcal{A}'$ is obtained from $\mathcal{A}$ by the following modifications:

1- the set of states Q and pushdown alphabet Γ are transformed into:

$$Q' := \{q_0\} \cup (Q \times \{0\}) \cup (Q \times \{1\})$$

$$\Gamma' := \{A_0'\} \cup \Gamma.$$

[2] the original formulation was slightly different since the constant series 1 of our statement was replaced by any rational *bounded* series

We interpret the second component of a state (q, i) as follows:
- $i = 0$ means that the automaton $\mathcal{A}$ has not yet read any letter on the input tape;
- $i = 1$ means that the automaton $\mathcal{A}$ has already read a symbol z on the input tape.
2- the transition map δ' of $\mathcal{A}'$ is defined by:

$$\begin{array}{rclll}
\delta(q_0, \varepsilon, A_0'\bar{y}) := & ((q_0, 0), \mathrm{push}_1(A_0A_0')) & \text{for } \bar{y} \in Y \cup \{\varepsilon\}, & (45)\\
\delta((q, i), \varepsilon, \gamma) := & ((r, i), \mathrm{op}) & \text{if } \delta(q, \varepsilon, \gamma) = (r, \mathrm{op}) & (46)\\
\delta((q, 0), \varepsilon, \gamma) := & ((r, 1), \mathrm{op}) & \text{if } \delta(q, z, \gamma) = (r, \mathrm{op}) & (47)\\
\delta((q, 1), z, \gamma) := & ((r, 1), \mathrm{op}) & \text{if } \delta(q, z, \gamma) = (r, \mathrm{op}) & (48)\\
\delta((q_0, i), \varepsilon, A_0'\bar{y}) := & (q_0, \mathrm{pop}_1) & \text{for } \bar{y} \in Y \cup \{\varepsilon\}, & (49)
\end{array}$$

Let $u \in Y^*$. If $(q_0, \varepsilon, A_0[u]) \vdash_{\mathcal{A}}^* (q_0, \varepsilon, \varepsilon)$ then

$$\begin{array}{rcll}
(q_0, \varepsilon, A_0'[u]) & \vdash_{\mathcal{A}'} & ((q_0, 0), \varepsilon, A_0[u]A_0'[u]) & \text{by transition (45)}\\
& \vdash_{\mathcal{A}'}^* & ((q_0, 0), \varepsilon, A_0'[u]) & \text{by transitions (46)}\\
& \vdash_{\mathcal{A}'} & ((q_0, \varepsilon, \varepsilon) & \text{by transition (49).}
\end{array}$$

Otherwise, if

$$(q_0, z^{n+1}, A_0[u]) \vdash_{\mathcal{A}}^* ((q_0, \varepsilon, \varepsilon)$$

then

$$\begin{array}{rcll}
(q_0, z^{n+1}, A_0'[u]) & \vdash_{\mathcal{A}'} & (q_0, 0), \varepsilon, A_0[u]A_0'[u]) & \text{by transition (45)}\\
& \vdash_{\mathcal{A}'}^* & ((q_0, 1), \varepsilon, A_0'[u]) & \text{by transitions } (47-48)\\
& \vdash_{\mathcal{A}'} & ((q_0, \varepsilon, \varepsilon) & \text{by transition (49).}
\end{array}$$

Finally, for all $u \in Y^*$

$$(q_0, z^{\max\{g(u)-1,0\}}, A_0'[u]) \vdash_{\mathcal{A}'}^* (q_0, \varepsilon, \varepsilon).$$

Since the map $u \mapsto z^{\max\{g(u)-1,0\}}$ is computable by a 2-sdpda, by Theorem 26 it is a HDT0L, i.e. since the image alphabet is a singleton, a $\mathbb{N}$-rational series. □

Theorem 44 *Let X, Y, Z, T be four finite alphabets and $f : X^* \to Y^*, g : Y^* \to Z^*, h : Z^* \to T^*$, be some maps. If f, h are rational maps and g is an HDT0L, then $f \circ g \circ h$ is an HDT0L.*

Proof: Every rational map is the composition of a left-sequential map with a right-sequential map (Theorem 5.2 of [Ber79]). We are thus reduced to prove Theorem 44 in the particular cases where h is the identity and f is right or left sequential and where f is the identity and h is right or left sequential. According to Definition 23, there exists an intermediate finite alphabet A, a homomorphism $H : Y^* \to \mathrm{HOM}(A^*, A^*)$, an homomorphism $h_0 \in \mathrm{HOM}(A^*, Z^*)$ and a letter $a_0 \in A$ such that, for every $w \in Y^*$

$$g(w) = h_0(H^w(a_0)).$$

Case 1:f is right-sequential, $Z = T$ and $h = \mathrm{Id}_{Z^*}$.
This case has been treated in Proposition 24.
Case 2:f is left-sequential, $Z = T$ and $h = \mathrm{Id}_{Z^*}$.
The map f is computed by some gsm

$$\mathcal{A} := \langle X, Y, Q, q_-, Q_+, \delta \rangle.$$

We borrow to the proof of Proposition 24 the operation $\otimes : Q \times A^* \to (Q \times A)^*$ (see formula (3)). We define below an homomorphism $K : X^* \to \mathrm{HOM}((Q \times A)^*, (Q \times A)^*)$ such that, for every $v \in A^*$,

$$K^u(q \otimes v) = (q \odot u) \otimes H^{q*u}(v) \tag{50}$$

It suffices to define K on each letter $x \in X$ by: for every $q \in Q, a \in A$

$$K^x(q, a) = (q \odot a) \otimes H^{q*x}(a).$$

Using (50) we get:

$$K^u(q_-, a_0) = (q_- \odot u) \otimes H^{q_- * u}(a_0) \tag{51}$$

Let us define an homomorphism $k_0 : (Q \times A)^* \to Z^*$ by: for every $q \in Q, a \in A$

$$k_0(q, a) := a.$$

From (51) and the definition of k_0 we get:

$$k_0(K^u(q_-, a_0)) = H^{f(u)}(a_0) = g(f(u))$$

which shows that $f \circ g$ is an HDT0L.
Case 3: $X = Y, f = \mathrm{Id}_{X^*}$ and h is left-sequential.
By Theorem 26 g is computed by some 2-sdpda $\mathcal{A} = (Q_{\mathcal{A}}, Z, \Gamma, \delta_{\mathcal{A}}, q_0, A_0)$ (where $Y \subseteq \Gamma$). Let $\mathcal{B} := \langle Z, T, Q_{\mathcal{B}}, q_{\mathcal{B},-}, Q_{\mathcal{B},+}, \delta_{\mathcal{B}} \rangle$ be a left-gsm that computes the map h.
Let us construct an automaton $\mathcal{C} = (Q, T, \Gamma, \delta_{\mathcal{C}}, (q_0, q_{\mathcal{B},-}), A_0)$ that is a kind of product of the sdpda $\mathcal{A}$ by the gsm $\mathcal{B}$:

$$\begin{aligned} Q &:= Q_{\mathcal{A}} \times Q_{\mathcal{B}} \\ \delta_{\mathcal{C}} &:= \{((q, r), \varepsilon, \gamma, (q', r), op) \mid (q, \varepsilon, \gamma, q', op) \in \delta_{\mathcal{A}}\} \\ &\quad \cup \{((q, r), u, \gamma, (q', r'), op) \mid \exists z \in Z, (q, z, \gamma, q', op) \in \delta_{\mathcal{A}}, r \odot z = r', r * z = u\} \end{aligned}$$

This automaton is not really a 2-sdpda in the sense of definition (11) because some of its transitions might read words u which have a length ≥ 2. Nevertheless, by a routine transformation, it can be translated into a 2-sdpda which computes the same map as $\mathcal{C}$, i.e. the map $g \circ h$.
Case 4: $X = Y, f = \mathrm{Id}_{X^*}$ and h is right-sequential.
By Theorem 26 g is a component g_{i_0} of a family $(g_i)_{i \in I}$ of maps, $g_i : Y^* \to Z^*$ (for $i \in I$), fulfilling a system of catenative recurrent relations:

$$g_i(yw) = \prod_{j=1}^{\ell(i,y)} g_{\alpha(i,y,j)}(w) \text{ for all } i \in I, y \in Y, w \in Y^* \tag{52}$$

where $\ell(y, i) \in \mathbb{N}, \alpha(i, y, j) \in I$. Let

$$\mathcal{B} := \langle Z, T, Q_{\mathcal{B}}, Q_{\mathcal{B}}, q_{\mathcal{B},+}, \delta_{\mathcal{B}} \rangle,$$

be a right-gsm that computes the map h. For every state $q \in Q_{\mathcal{B}}$, we denote by $h_q : Z^* \to T^*$ the right-sequential map computed by the right-gsm

$$\mathcal{B}_q := \langle Z, T, Q_{\mathcal{B}}, Q_{\mathcal{B}}, q, \delta_{\mathcal{B}} \rangle.$$

Let us denote by $\approx$ the equivalence relation over Z^* defined by: for every $w, w' \in Z^*$

$$w \approx w' \Leftrightarrow \forall q \in Q_{\mathcal{B}}, w \odot q = w' \odot q.$$

It is well known that $\approx$ is a congruence of finite index. Let us define a binary relation $\equiv$ over Y^* by: for every $v, v' \in Y^*$

$$v \equiv v' \Leftrightarrow \forall i \in I, g_i(v) \approx g_i(v').$$

We claim that $\equiv$ is a left-congruence of finite index.
Suppose that

$$\forall i \in I, g_i(v) \approx g_i(v'). \tag{53}$$

Then, by the recurrence relations (52), for every $i \in I, y \in Y$,

$$g_i(yv) = \prod_{j=1}^{\ell(i,y)} g_{\alpha(i,y,j)}(v) \approx \prod_{j=1}^{\ell(i,y)} g_{\alpha(i,y,j)}(v') \approx g_i(yv'). \tag{54}$$

The implication (53) $\Rightarrow$ (54) shows that $\equiv$ is left-compatible with product. Concerning the index, it is clear that

$$\mathrm{Card}(Y^*/\equiv) \leq \mathrm{Card}(Z^*/\approx)^{\mathrm{Card}(I)}.$$

Let us define, with the help of this left-congruence $\equiv$, a system of left-regular catenative recurrent relations over the family $(g_i \circ h_q)_{i \in I, q \in Q_{\mathcal{B}}}$. We first define a sequence of states of $\mathcal{B}$ by:

$$
\begin{array}{rcl}
r(i, y, \ell(i, y), c, q) & := & q \\
r(i, y, j-1, c, q) & := & g_{\alpha(i,y,j)}(v) \odot r(i, y, j, c, q) \\
& & \text{for all } i \in I, y \in Y, c \in Y^*/\equiv, v \in c.
\end{array} \qquad (55)
$$

and then the catenative recurrence relations themselves:

$$
\begin{array}{rcl}
h_q(g_i(yv)) & = & \prod_{j=1}^{\ell(i,x)} h_{r(i,y,j,c,q)}(g_{\alpha(i,y,j)}(v)) \\
& & \text{for all } i \in I, y \in Y, c \in Y^*/\equiv, v \in c.
\end{array} \qquad (56)
$$

By Lemma 27, every left-regular catenative recurrent sequence is a HDT0L, hence $g \circ h = g_{i_0} \circ h_{q_+}$ is an HDT0L. □
Note that the skimming theorem corresponds to the situation where f is the identity map and $h : z^* \to z^*$ is defined by $h(z^n) := z^{\max\{0,n-1\}}$.
The situation where h is the identity map over $\{z\}^*$, $X = Y$ and $f : X^* \to X^*$ is defined by $f(u) = xu$ (for some letter $x \in X$), corresponds to the (classical) statement that every residual of a rational series is a rational series.

5.2 Rational sets of homomorphisms

The notion of rational subset of a monoid $\mathbb{M}$ can be defined in general: $\mathrm{RAT}(\mathbb{M})$ is the least element of $\mathcal{P}(\mathcal{P}(\mathbb{M}))$ such that:

- for every $m \in \mathbb{M}$, $\{m\} \in \mathrm{RAT}(\mathbb{M})$
- $\mathrm{RAT}(\mathbb{M})$ is closed under union, product and star

(we recall the star-operation maps every subset $P \in \mathcal{P}(\mathbb{M})$) to the least submonoid containing P (denoted by P^*)). It is well-known that, in the particular case where $\mathbb{M}$ is a free monoid or a free group ([Ber79]) or a commutative monoid ([ES69]), $\mathrm{RAT}(\mathbb{M})$ is closed under intersection. We show here that the monoid $\mathrm{HOM}(A^*, A^*)$ does not have this property: we exhibit two rational subsets R, S of $\mathrm{HOM}(A^*, A^*)$ such that $R \cap S$ is not rational. In order to prove that $R \cap S$ is not rational we exploit the link between homomorphisms and pda of level 2 that is given by Theorem 26, together with some already known property of indexed-languages (i.e. languages recognized by pda of level 2).
Let us denote by $\diamond$ the right-action of homomorphisms over A^* i.e.

$$\forall f \in \mathrm{HOM}(A^*, A^*), \forall w \in A^*, w \diamond f := f(w).$$

Let us analyze the result of a rational set of homomorphisms acting on a fixed word.

Lemma 45 *Let $a \in A$. Let X be some finite alphabet and $H : X^* \to \mathrm{HOM}(A^*, A^*)$ be a monoid homomorphism. The set $\{H^u(a) \mid u \in X^*\}$ is an indexed-language.*

Sketch of proof: By Theorem 26, since $u \mapsto H^u(a)$ is an HDT0L, it is computed by some sdpda $\mathcal{A} = (Q, A, \Gamma, \delta, q_0, A_0)$ such that $X \subseteq \Gamma$. Let us consider a non-deterministic pda $\mathcal{A}'$ of level 2 which behaves as follows: on every input $v \in A^*$, it guesses (by a sequence of non-deterministic transitions), a word $u \in X^*$ that it pushes into its first 1-dimensional pushdown:

$$(q_0', v, A_0'[\varepsilon]) \vdash_{\mathcal{A}'}^* (q_0', v, A_0'[u]). \tag{57}$$

In a second step, it moves to a starting configutation of $\mathcal{A}$

$$(q_0', v, A_0'[u]) \vdash_{\mathcal{A}}^* (q_0, v, A_0[u]). \tag{58}$$

and checks, by the unique maximal computation of $\mathcal{A}$, whether $H^u(a) = v$:

$$(q_0, v, A_0[u]) \vdash_{\mathcal{A}}^* (q, \varepsilon, \varepsilon). \tag{59}$$

It should be clear that one can define a 2-pda $\mathcal{A}'$ with this behavior. This automaton $\mathcal{A}'$ recognizes the set $\{H^u(a) \mid u \in X^*\}$. $\square$

Lemma 46 *For every rational subset R of $\mathrm{HOM}(A^*, A^*)$ and every $w \in A^*$, the set $\{w \diamond f \mid f \in R\}$ is an indexed-language.*

Proof: The strategy of this proof is to describe $L := \{w \diamond f \mid f \in R\}$ in a way very close to an HDT0L and to exploit this similarity.
Let us fix $w \in A^*$ and choose some letter $a_0 \in A$ (which will serve as initial letter for some HDT0L). Since R is rational, there exists some finite alphabet X, a monoid homomorphism $H : X^* \to \mathrm{HOM}(A^*, A^*)$ and a rational subset $R_X \subseteq X^*$ such that

$$R = \{H^u \mid u \in R_X\}.$$

The language L can be rewritten as

$$L = \{H^u(w) \mid u \in R_X\}.$$

We extend the alphabets X, A into

$$X' := X \cup \{b, e, f\}$$

where b, e, f, are fresh symbols (distinct and not belonging to X)

$$A' := A \cup \bar{A}$$

where $\bar{A}$ is a new alphabet, endowed with a bijection $a \mapsto \bar{a}$ from A to $\bar{A}$, and disjoint from A. The homorphism H is extended as a monoid homomorphism $H : X' \to \mathrm{HOM}(A'^*, A'^*)$ by:

$$H^b: \quad a_0 \mapsto w, \;\; a \mapsto a \text{ if } a \in A' \setminus \{a_0\}$$
$$H^e: \quad a \mapsto a \text{ for all } a \in A'$$
$$H^f: a \mapsto \bar{a} \text{ for all } a \in A, \;\; \bar{a} \mapsto a \text{ for all } a \in A, \tag{60}$$

and we define a rational map $h : X^* \to X'^*$ by

$$h : u \mapsto bue \text{ if } u \in R_X, \;\; u \mapsto buf \text{ if } u \in X^* \setminus R_X. \tag{61}$$

One can check that, for every $u \in X^*$:

$$H^{h(u)}(a_0) = H^u(w) \text{ if } u \in R_X, \;\; H^{h(u)}(a_0) = \overline{H^u(w)} \text{ if } u \in X^* \setminus R_X.$$

It follows that

$$L = \{H^{h(u)}(a_0) \mid u \in X^*\} \cap A^*. \tag{62}$$

By Theorem 44 the map $u \mapsto H^{h(u)}(a_0)$ is an HDT0L. Hence, by Lemma 45 $\{H^{h(u)}(a_0) \mid u \in X^*\}$ is an indexed language and, using (62), L is an indexed language too (because the family of indexed-languages is closed under intersection with rational sets). □

Let us consider the alphabet

$$A := \{x, y, z, t\}$$

and the following elements h_1, h_2, h_3 of $\mathrm{HOM}(A^*, A^*)$:

$$\begin{array}{lllll} h_1: & x \mapsto xy, & y \mapsto y, & z \mapsto z, & t \mapsto t, \\ h_2: & x \mapsto x, & y \mapsto yx, & z \mapsto z, & t \mapsto t, \\ h_3: & x \mapsto x, & y \mapsto y, & z \mapsto zt, & t \mapsto t, \end{array} \tag{63}$$

Let

$$R := (h_1 \circ h_3)^* \circ h_2^*, \;\; S := h_1^* \circ (h_2 \circ h_3)^*$$

Proposition 47 *R and S are rational subsets of* $\mathrm{HOM}(A^*, A^*)$ *while $R \cap S$ is not rational.*

Proof: Let us first observe that $h_1 \circ h_3 = h_3 \circ h_1$ and $h_2 \circ h_3 = h_3 \circ h_2$. Hence

$$R = \{h_1^n h_2^m h_3^n \mid n \geq 0, m \geq 0\}, \;\; S = \{h_1^n h_2^m h_3^m \mid n \geq 0, m \geq 0\}.$$

Let $h \in R \cap S$: there exist $n, m, p, q \in \mathbb{N}$ such that

$$h = h_1^n h_2^m h_3^n = h_1^p h_2^q h_3^q$$

But, $z \diamond h = zt^n = zt^q$. Hence $n = q$.
And $x \diamond h = x(yx^m)^n = x(yx^q)^p$. Hence $n = p$ and $m = q$. Finally, $n = m = p = q$ so that $h = h_1^n h_2^n h_3^n$. From this follows the equality:

$$R \cap S = \{h_1^n h_2^n h_3^n \mid n \geq 0\}.$$

It follows that

$$\{x \diamond h \mid h \in R \cap S\} = \{x(yx^n)^n \mid n \geq 0\}.$$

But this last language has been shown to do not be an indexed-language in ([Hay73], Th. 5.3) and ([Gil96], Corollary 4). By Lemma 46, we can conclude that $R \cap S$ is not rational. □

5.3 Set of solutions of a quadratic equation

The celebrated theorem of Makanin ([Mak77],[Die02]) establishes that the satisfiability problem for finite systems of equations (with constants) over words, is decidable. A description of the full set of solutions of a finite system of word-equations is given in [Pla06]. Nevertheless the type of description that is used in [Pla06] is not well-related to the classical notions of automata and grammars. We show here that, in the case of *quadratic* equations, the set of all solutions is an *indexed-language* i.e. is recognized by some pda of level 2 (this result was first proved in [Gue07], by a more direct construction).

Let $\mathcal{U}$ be some finite alphabet. A *system of word equations*, with set of variables $\mathcal{U}$, is a subset $\mathcal{S} \subseteq \mathcal{U}^* \times \mathcal{U}^*$. A *solution* of the system $\mathcal{S}$ is any monoid homomorphism

$$\sigma : \mathcal{U}^* \to \mathcal{U}^*$$

fulfilling:

$$\forall (w, w') \in \mathcal{S} : \quad \sigma(w) = \sigma(w') \tag{64}$$

It is clear, from the proof given in [Die02], that the set of solutions, viewed as homorphisms

$$\mathrm{SOL}_h(\mathcal{S}) := \{\varphi \in \mathrm{HOM}(\mathcal{U}^*,\mathcal{U}^*) \mid \forall (w, w') \in \mathcal{S}, \varphi(w) = \varphi(w')\}$$

is a rational subset of $\mathrm{HOM}(\mathcal{U}^*,\mathcal{U}^*)$ composed (on the right) by the full set $\mathrm{HOM}(\mathcal{U}^*,\mathcal{U}^*)$ i.e. there exists a rational subset $R \subseteq \mathrm{HOM}(\mathcal{U}^*,\mathcal{U}^*)$ such that

$$\mathrm{SOL}_h(\mathcal{S}) = R \circ \mathrm{HOM}(\mathcal{U}^*,\mathcal{U}^*). \tag{65}$$

Let us order the elements of $\mathcal{U}$ as $u_0, u_1, \ldots, u_{n-1}$ and represent the solutions as words:

$$\mathrm{SOL}_w(\mathcal{S}) := \{\varphi(u_0)\#\varphi(u_1)\# \cdots \#\varphi(u_{n-1}) \mid \varphi \in \mathrm{SOL}_h\}.$$

The set of solutions is thus viewed as a subset of $(\mathcal{U} \cup \{\#\})^*$.

Theorem 48 *For every finite quadratic system of word equations $\mathcal{S}$, the set of word-solutions* $\mathrm{SOL}_w(\mathcal{S})$ *is an indexed-language.*

Proof: Let $w_0 := u_0\#u_1\#\cdots\#u_{n-1}$. We would like to express the set $\mathrm{SOL}_w(\mathcal{S})$ under the form

$$\{w_0 \diamond f \mid f \in S\}$$

for some rational subset S of $\mathrm{HOM}(\mathcal{U}^*,\mathcal{U}^*)$. Unfortunately, it is not obvious that equation (65) furnishes such a ratiuonal set S, just because we do not know whether $\mathrm{HOM}(\mathcal{U}^*,\mathcal{U}^*)$ is a rational subset of itself (i.e. is finitely generated). We shall circunvent this technical difficulty by considering a larger alphabet. Let $\bar{\mathcal{U}}$ be a new alphabet in one to one correspondance with $\mathcal{U}$ and such that $\bar{\mathcal{U}} \cap \mathcal{U} \cap \{\#\} = \emptyset$. We define an extended alphabet

$$\mathcal{U}' := \mathcal{U} \cup \bar{\mathcal{U}} \cup \{\#\}.$$

For every homomorphism $h \in \mathrm{HOM}(\mathcal{U}^*,\mathcal{U}^*)$, and every letters $u, v \in \mathcal{U}$ we introduce the following homomorphisms from $\mathcal{U}'^*$ into itself:

$$\begin{aligned}
\alpha &: u' \mapsto \bar{u'}(\forall u' \in \mathcal{U}),\ \ \bar{u'} \mapsto u'(\forall u' \in \mathcal{U}),\ \ \# \mapsto \#,\\
f_{u,v} &: u' \mapsto u'(\forall u' \in \mathcal{U}),\ \ \bar{v} \mapsto u\bar{v},\ \ \bar{v'} \mapsto \bar{v'}(\forall v' \in \mathcal{U} \setminus \{v\}),\ \ \# \mapsto \#,\\
\bar{f}_v &: u' \mapsto u'(\forall u' \in \mathcal{U}),\ \ \bar{v} \mapsto \varepsilon,\ \ \bar{v'} \mapsto \bar{v'}(\forall v' \in \mathcal{U} \setminus \{v\}),\ \ \# \mapsto \#,\\
h_v &: u' \mapsto u'(\forall u' \in \mathcal{U}),\ \ \bar{v} \mapsto h(v),\ \ \bar{v'} \mapsto \bar{v'}(\forall v' \in \mathcal{U} \setminus \{v\}),\ \ \# \mapsto \#,
\end{aligned}$$

One can check that

$$h_v \in \{f_{u,v} \mid u \in \mathcal{U}\}^* \circ \bar{f}_v$$

and, for every letter $x \in \mathcal{U} \cup \{\#\}$,

$$x \diamond h = x \diamond (\alpha \circ \prod_{v \in \mathcal{U}} h_v)$$

Let us denote by $\hat{R}, \hat{S}$ the following rational subsets of $\mathrm{HOM}(\mathcal{U}'^*,\mathcal{U}'^*)$:

$$\hat{R} := \{\hat{h} \in \mathrm{HOM}(\mathcal{U}'^*,\mathcal{U}'^*) \mid \exists h \in R, \hat{h}_{|\mathcal{U}^*} = h, \forall w \in (\bar{\mathcal{U}} \cup \{\#\})^* \hat{h}(w) = w\}$$

(since R is rational, $\hat{R}$ is a rational subset of $\mathrm{HOM}(\mathcal{U}'^*,\mathcal{U}'^*)$ too)

$$\hat{S} := \alpha \circ \prod_{v \in \mathcal{U}} \{f_{u,v} \mid u \in \mathcal{U}\}^* \circ \bar{f}_v$$

Using the previous description of the set of solutions (65) we obtain the following description of the set of word-solutions:

$$\mathrm{SOL}_w(\mathcal{S}) = \{w_0 \diamond f \mid f \in \hat{R} \circ \hat{S}\}$$

Since $\hat{R} \circ \hat{S}$ is rational, by Lemma 46, $\mathrm{SOL}_w(\mathcal{S})$ is an indexed language. □
Let us remark that, since the family of indexed-languages is closed under intersection with rational sets, Theorem 48 remains true for finite quadratic systems of equations with *rational constraints* (or, in particular, with *constants*).

6 Related works

- In [ERS80] many links between Lindenmayer systems and transducers are demonstrated;
- The work [EV86] does prove a statement which is rather close to our main theorem, though, since it deals with terms and strings (and not exclusively with strings) it uses a different terminology:
  - the mappings from strings to strings corresponding to our notion of recurrent catenative sequences coincide (up to some normalisation) with the Deterministic Context-Free-grammars with indexes in the structure TR; they are denoted by $\mathrm{D}_t\,\mathrm{CF}(\mathrm{TR})$
  - the mappings from strings to strings corresponding to our notion of sequences computed by sdpda of level 2, but without any use of the operation push_2 , are called deterministic REGular grammars with indexes in the structure P(TR); they are denoted by $\mathrm{D}_t\,\mathrm{REG}(\mathrm{P}(\mathrm{TR}))$.
  Our main theorem, together with our normalisation Lemma 42 implies, in terms of this formalism, that
  $$\mathrm{D}_t\,\mathrm{CF}(\mathrm{TR}) = \mathrm{D}_t\,\mathrm{REG}(\mathrm{P}(\mathrm{TR})) \tag{66}$$
  in the case of strings i.e. of a graded alphabet where all the operation symbols have arity 1 or 0. The above equality was also a consequence of Theorem 6.7 p. 335 of [EV86] asserting that, for every storage structure S,
  $$\text{if } S_{LA} \equiv S, \text{ then } \mathrm{D}_t\,\mathrm{CF}(S) = \mathrm{D}_t\,\mathrm{REG}(\mathrm{P}(S)) \tag{67}$$
  once we know that the storage structure TR, *in the particular case of symbols of arity* ≤ 1, does fulfill the hypothesis $S_{LA} \equiv S$.
  Note that for the general storage structure TR (i.e with operation symbols of arbitrary arity), the equality (66) does not hold any more (see the remark on line 1, p.336 of [EV86], refering to [ERS80]). Hence, to resume our comparison:
  - property (66) for strings is both a corollary of our main theorem and also of Theorem 6.7 p. 335 of [EV86]
  - our main theorem improves, in the case of strings, property (66), since it holds without any restriction on the push_2 moves.
- In [LK03] some classes of maps $\mathbb{N} \to \mathbb{N}$ are defined through index-grammars. They are carefully examined and, for example, one class (named IGF_0),

is shown to coincide with the class of integer sequences which are strictly increasing and fulfill some linear recurrence with natural integer coefficients;

- In [Sén07] the notion of mapping from strings to strings computed by sdpda is extended to all levels $k \geq 3$ of the hierarchy of pushdown automata; a characterisation of $\mathbb{S}_3(X^*, \mathbb{N})$ as the sequences defined by *polynomial* recurrence relations is proved.
  It is announced in Theorem 6 of ([Sén07]) that the mappings of level $k \geq 2$ are exactly the $(k-1)$-fold compositions of mappings of level 2, thus motivating the present careful study of level 2.
- L. Braud examined the structure of infinite words that are recognized by $k-$dpda ([Bra10], chapter 5; [BC10, Theorem 2]); some links between the case of *infinite* words and the case of *sequences of finite* words (which are the main concern of the present paper) can be expected.

6.0.0.4 Aknowledgments This work has been partly supported by the project ANR 2010 BLAN 0202 01 FREC.

References

[BC10] L. Braud and A. Carayol. Linear orders in the pushdown hierarchy. In *Proceedings ICALP'10*, volume 6199 of LNCS, pages 88–99. Springer-Verlag, 2010.

[Ber79] J. Berstel. *Transductions and contect-free languages*. Teubner, 1979.

[BO93] R.V. Book and F. Otto. *String Rewriting Systems.Texts and monographs in Computer Science*. Springer-Verlag, 1993.

[Bra10] L. Braud. *The structure of order in the pushdown hierarchy .Les structures d'ordre dans la hiérarchie à pile*. Thèse de doctorat, Université Paris-Est, Juin 2010. Directeur: D. Caucal.

[DG86] W. Damm and A. Goerdt. An automata-theoretical characterization of the OI-hierarchy. *Information and control*, 71:1–32, 1986.

[Die02] V. Diekert. Makanin's algorithm. In *Algebraic combinatorics on words, Chap. 12*, pages 387–438. Cambridge University Press, Cambridge, U.K., 2002.

[ERS80] Joost Engelfriet, Grzegorz Rozenberg, and Giora Slutzki. Tree transducers, L systems, and two-way machines. *J. Comput. System Sci.*, 20(2):150–202, 1980. ACM-SIGACT Symposium on the Theory of Computing (San Diego, Calif., 1978).

[ES69] Samuel Eilenberg and M. P. Schützenberger. Rational sets in commutative monoids. *J. Algebra*, 13:173–191, 1969.

[EV86] J. Engelfriet and H. Vogler. Pushdown machines for the macro tree transducers. *Theoretical Computer Science*, 42:251–368, 1986.

[FS06] S. Fratani and G. Sénizergues. Iterated pushdown automata and sequences of rational numbers. *Ann. Pure Appl. Logic*, 141(3):363–411, 2006.

[Gil96] Robert H. Gilman. A shrinking lemma for indexed languages. *Theoret. Comput. Sci.*, 163(1-2):277–281, 1996.

[Gue07] M. Guedj. Equations et langages formels, le cas quadratique. *Master Thesis, Bordeaux 1 university*, 2007.

[Hay73] T. Hayashi. On derivation tress of index grammars: An extension of the uvwxy-theorem. *Publ. RIMS Kyoto Univ.*, 9:61–92, 1973.

[Hue80] G. Huet. Confluent reductions: abstract properties and applications to term rewriting systems. *JACM*, 27(4):797–821, 1980.

[KRS97] L. Kari, G. Rozenberg, and A. Salomaa. L systems. In *Handbook of formal languages, Vol. 1*, pages 253–328. Springer, Berlin, 1997.

[LK03] L. P. Lisovik and T. A. Karnaukh. On a class of functions computable by index grammars. *Kibernet. Sistem. Anal.*, (1):108–115, 191, 2003.

[Mak77] G. S. Makanin. The problem of the solvability of equations in a free semigroup. *Mat. Sb. (N.S.)*, 103(145)(2):147–236, 319, 1977.

[Mas74] A.N. Maslov. The hierarchiy of indexed languages of an arbitrary level. *Soviet. Math. Dokl. 15*, pages 1170–1174, 1974.

[Mas76] A.N. Maslov. Multilevel stack automata. *Problemy Peredachi Informatsii 12*, pages 38–43, 1976.

[Pla06] Wojciech Plandowski. An efficient algorithm for solving word equations (extended abstract). In *STOC'06: Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, pages 467–476. ACM, New York, 2006.

[Sak03] J. Sakarovitch. *Éléments de théorie des automates*. Vuibert Informatique, 2003.

[Sch70] M.-P. Schützenberger. Parties rationnelles d'un monoïde libre. *Congrès International des Mathématiciens*, 3:281–282, 1970.

[SdS08] J. Sakarovitch and R. de Souza. On the decomposition of k-valued rational relations. In *Proceedings of STACS 2008*, pages 621–632, 2008.

[Sén07] G. Sénizergues. Sequences of level 1,2,3,...,k,... In *Proceedings CSR'07*, volume 4649 of LNCS, pages 24–32. Springer-Verlag, 2007. Invited talk at CSR'07.