LaBRI, CNRS UMR 5800
Laboratoire Bordelais de Recherche en Informatique

Rapport de recherche RR-1457-12

On languages of one-dimensional overlapping tiles

April 10, 2012

David Janin,
LaBRI, IPB, Université de Bordeaux

Abstract

A one-dimensional tile with overlaps is a standard finite word that carries some more information that is used to say when the concatenation of two tiles is legal. Known since the mid 70's [11] in the rich mathematical field of inverse monoid theory, this model of tiles with the associated partial product have yet not been much studied in theoretical computer science despite some implicit appearances in studies of two-way automata in the 80's [17, 3].

We aim in this paper at initializing such a systematic computer science flavored theoretical study. For that purpose, after describing the richness of the underlying algebraic structure, we define and study several classical classes of languages of tiles: from recognizable languages definable by morphism into finite monoids up to languages definable in monadic second order logic (MSO).

We show that recognizable languages of tiles are tightly linked with covers of periodic bi-infinite words. We also show that the class of MSO definable languages of tiles is both *simple*: these languages are finite sums of Cartesian products of rational languages, and *robust*: the class is closed under product, iterated product (star) and shifts (two tiles specific operators). An equivalent notion of regular expression is then provided.

Introduction

In this paper, we study languages of one-dimensional discrete overlapping tiles.

These tiles already appear in the 70's as elements of McAlister monoid [11] in the rich mathematical field of inverse monoid theory [10]. In particular, though sometimes implicitly, they are used for studying the structure of (zigzag) covers of finite, infinite or bi-infinite words [12, 1]. They also appear in studies of the structure of tiling (in the usual sense with no overlaps) of the d-dimensional Euclidian space $\mathbb{R}^d$ [8, 1].

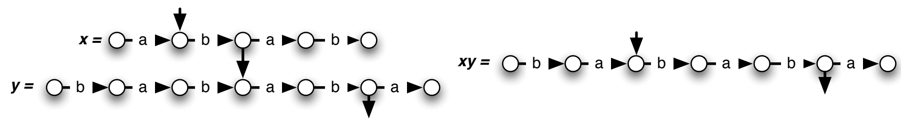
Following the cover point of view, overlapping tiles can be defined as triples of finite words of the form (u, v, w) where v defines *what* part is primarily covered by the tile while u and w define left and right matching constraints that tell more *where* such a tile can be used.

For instance, covering the bi-infinite words ${}^{\omega}(ab).(ab)^{\omega} = \cdots abababab \cdots$ allowed tiles are all and nothing but tiles of the form (u, v, w) such that the word uvw is a factor of the bi-infinite word ${}^{\omega}(ab).(ab)^{\omega}$, i.e. $uvw \in a(ba)^* + (ba)^* + (ab)^* + b(ab)^*$.

In such a tile (u, v, w), the word v, called the *root* of the tile, must be a factor of the word to be covered, and, both u and w, respectively called the *left and right matching constraints* of that tile, must also match the left and right neighborhood of the factor covered by word v.

In such a cover, two neighbor tiles can then be seen as a single larger tile: the concatenation product of the two tiles. Although a little hard to understand at first sight, the product of two tiles is just defined as the concatenation of their roots, the resulting left and right matching constraints being checked and defined accordingly.

For instance, given two tiles $x = (a, b, ab)$ and $y = (bab, ab, a)$ we have $xy = (ba, bab, a)$.



In this example, the resulting left constraint ba of xy is actually inherited from the left constraint bab of y by canceling from the right, the root b of tile x. With the same tiles, the product yx is undefined. This is because, in particular, the root b of x does not match the right constraint a of y. This fact is conveniently modeled by taking $yx = 0$ where 0 is an extra additional tile seen as the *undefined tile*.

The resulting algebraic structure is a monoid: the monoid of positive tiles. We aim in this paper to study this monoid and its associated language theory.

Main results

We first describe the rich though basic properties of the monoid of positive tiles. Two extra operators, the left and right shifts of positive tiles, are added and studied. This addition is the price to pay for obtaining a finitely generated algebraic structure.

Completing the monoid of positive tiles with negative tiles, we show that the monoid of positive tiles is a submonoid of McAlister inverse monoid [11] (see Theorem 7). Left and right shift operators are shown to be definable by means of inverses.

Three classes of languages of positive tiles are then considered : the class REC of languages definable as inverse images of (safe) monoid morphisms into finite monoids, the class RAT_S of languages definable by means of finite rational expressions extended with left and right shifts, and the class MSO of languages of tiles definable by means of a formulae of monadic second order logic.

The class REC, strictly included into MSO, is shown to be tightly linked with covers of periodic bi-infinite words (see Theorem 21). On the opposite side, the largest class MSO is shown to be both *robust*: $RAT_S \subseteq MSO$ (see Theorem 13) and *simple*: these languages are finite sums of Cartesian products of rational languages of words (see Theorem 12). The reverse inclusion $MSO \subseteq RAT_S$ follows from that latter fact.

Altogether, we prove that $REC \subset RAT_S = MSO$.

Related works

The monoid of positive and negative tiles, defined and studied in this paper, has already been considered in the 70's in inverse monoid theory as the monoid of McAlister. In [10], Lawson proposes a alternative presentation of that monoid in Chapter 9.4. As far we know, our presentation by means of triples of words, more symmetric, is original. The proof of equivalence with McAlister monoid, directly via Munn's birooted Trees [15], puts the emphasis on the fact that non zero tiles coincide with unidirectional and linear birooted trees.

In the 80's, tiles also implicitly appear in the work of Pécuchet on two-way automata [17]. Tiles are just partial runs of two-way automata (pairs of sections in author's terminology). A connection with inverse or regular monoid is made, but the connection with McAlister monoid itself is at best left implicit. Still related with two-way automata, Birget approach in the late 80's [3] also makes use of algebraic structures that sounds like monoid of tiles : an approach that has been used till recently [9]. We believe that a careful study of two-way automata in relationship with McAlister monoid could leads to interested development. But such a study still need to be done.

Last, we must say that in a companion paper [7], we investigate a remedy to the collapse of algebraic recognizability by means weakening the notion of monoid morphism to the notion of McAlister's and Reilly's prehomomorphism [10]. There, we essentially achieve to obtain an algebraic characterization of MSO definable languages of tiles. This study also led to a generalization of the monoid of positive tiles, in the border of inverse monoid theory, that have been studied independently [6].

Oddly enough, our interest in studying languages of positive tiles came from application perspectives in computational music theory. The purpose of the present paper is however and by no means to defend such a point of view. This would require, to be convincing, many more than few lines. We refer the interested reader to [5][1].

Some notations

Given a finite alphabet A, let A^* be the free monoid generated by A with neutral element denoted by 1 and let A^ω (resp. ${}^\omega\!A$) be the set of right infinite words (resp. left infinite words) on the alphabet A. The concatenation of two words u and v is denoted by $u.v$ or even simply uv.

For all non empty finite word $u \in A^*$, we denote by $u^\omega \in A^\omega$ (resp. ${}^\omega u \in {}^\omega\!A$) the right (resp. left) infinite words obtained by an infinite right (resp. left) repetition of the word u. In particular, we use the notation ${}^\omega u u^\omega$ for the bi-infinite word (with no origin) defined by the bi-infinite repetition of word u.

Let $\leq_p$ stands for the prefix order over $A^* + A^\omega$. For all word x and y, if $x \leq_p y$, let $x^{-1}(y)$ stands for the unique word such that $x.x^{-1}(y) = y$. Symmetrically, let $\leq_s$ stands for the suffix order over $A^* + {}^\omega\!A$. For all word x and y, if $x \leq_s y$, let $(y)x^{-1}$ stands for the unique word such that $(y)x^{-1}.x = y$.

Let also $\vee_p$ and $\wedge_p$ denote the joint and meet operators for the prefix order and $\vee_s$ and $\wedge_s$ denote the joint and meet operators for the suffix order. For all word u and v, $u \wedge_p v$ (resp. $u \wedge_s v$) is the greatest common prefix (resp. suffix) of the words u and v. Symmetrically, $u \vee_p v$ (resp. $u \vee_s v$) is the least word such that both u and v are prefix (resp. suffix) of that word.

[1] this research report is, unfortunately for most readers, in french; be sure we aim to report soon on that topic in english

The free monoid A^* extended with 0 is a bi-lattice. In particular, since $0.u = u.0 = v.0 = 0.v = 0$, one has $u \vee_p v = 0$ (resp. $u \vee_s v = 0$) whenever neither u is prefix (resp. suffix) of v nor v is prefix (resp. suffix) of u.

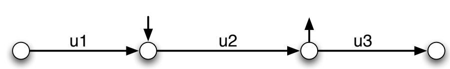
Given $\bar{A}$ a disjoint copy of A, we write also $u \mapsto \bar{u}$ for the mapping from $(A + \bar{A})^*$ to itself inductively defined by $\bar{1} = 1$, for all letter $a \in A$, $\bar{a}$ is the copy of a in $\bar{A}$, $\bar{\bar{a}} = a$ and, for all word $u \in (A + \bar{A})^*$, $\overline{au} = \bar{u}.\bar{a}$. The mapping $u \mapsto \bar{u}$ is an antimorphism, i.e. for all word u and $v \in (A + \bar{A})^*$, $\overline{uv} = \bar{v}.\bar{u}$ and an involution, i.e. for all word $u \in (A + \bar{A})^*$, $\bar{\bar{u}} = u$.

Last, the free group $FG(A)$ generated by A is defined to be $(A + \bar{A})^*$ quotiented by the least congruence over $(A + \bar{A})^*$ such that, for all letter $a \in A$, $a\bar{a} = 1$ and $\bar{a}a = 1$. For all $w \in (A + \bar{A})^*$ there is a unique reduced word $red(w)$ equivalent to w, such that there is no occurrence of $a\bar{a}$ nor $\bar{a}a$ in $red(w)$. In the remainder of the text, we shall identify elements of $FG(A)$ to their reduced. With that assumption, for all u and v in A^*, if $u \leq_p v$ then $u^{-1}(v) = \bar{u}v$ and if $u \leq_s v$ then $(v)u^{-1} = v\bar{u}$.

1 Monoid of positive tiles

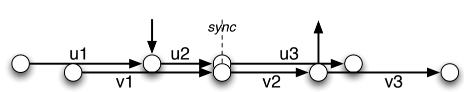
A positive (or left to right) tile over the alphabet A is define to be a triple of words $u = (u_1, u_2, u_3) \in A^* \times A^* \times A^*$. In such a tile $u = (u_1, u_2, u_3)$, word u_1 is called the *left constraint*, word u_2 is called the *root* and word u_3 is called the *right constraint* of tile u. The word $u_1u_2u_3$ is called the *domain* of tile u.

Tile $u = (u_1, u_2, u_3)$ is conveniently drawn as a (linear, unidirectional and left to right) Munn's birooted word tree [15]:



where a dangling input arrow marks the beginning of the root and a dangling output arrow marks the end of the root.

The sequential product of two non-zero tiles $u = (u_1, u_2, u_3)$ and $v = (v_1, v_2, v_3)$ is intuitively defined as the tile (if it exists) resulting of the superposition of the two tiles positioned in such a way that the end of the root of the first tile *coincides* with the beginning of the root of the second tile, with a *pattern-matching condition*, to the left and to the right of that synchronization point, that ensures that superposed letters are equal.



The root of the resulting product tile is then defined as the concatenation of the roots of the components of the product.

Formally, for any two positive tiles $u = (u_1, u_2, u_3)$ and $v = (v_1, v_2, v_3)$, the *product of u and v* is defined by

$$u.v = ((u_1.u_2 \vee_s v_1)u_2^{-1}, u_2v_2, v_2^{-1}(u_3 \vee_p v_2v_3))$$

when the *left matching constraint* $u_1.u_2 \vee_s v_1 \neq 0$, i.e. u_1u_2 and v_1 are suffix one of the other, and the *right matching constraint* $u_3 \vee_p v_2v_3 \neq 0$, i.e. u_3 and v_2v_3 are prefix one of the other, are both satisfied.

We complete the set of positive tiles by an extra tile, denoted by 0, called the *undefined tile*. The partial product of tiles is then completed as follows: for all non zero tile u and v, $u.v = 0$ when the matching constraints are not satisfied, and, for all tile u, by $u.0 = 0.u = 0$.

Remark. Let a, b, c and $d \in A$ be four distinct letters. Then $(a, b, c).(b, c, d) = (a, bc, d)$ while $(a, b, c).(a, c, d) = 0$. In the latter case, the left matching constraint is violated because $a \neq b$.

Theorem 1 *The set $T_A = A^* \times A^* \times A^* + 0$ equipped with the sequential product of tiles is a monoid with neutral element $1 = (1, 1, 1)$ and absorbing element 0.*

Proof. The fact that 1 is neutral is immediate. Associativity of the sequential product follows from the fact that the domain of a non-zero product always contains the domains of its components. □

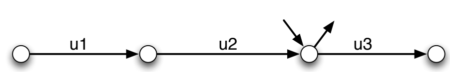
2 Shifts, context elements and natural order

In the concatenation product of tiles, roots are concatenated and thus behave like usual finite words. On the contrary, in a product, both left and right constraints decrease. It follows that while arbitrary tile's roots can be generated, via tiles product, from (canonical tiles built from) letters of A, this is not true for tile's constraints. In other words, monoid T_A is not finitely generated.

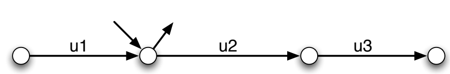
The left and right shift operators, defined below, give a solution for generating arbitrary tiles from (canonical images of) words of A^*. They also induce a notion of context elements that leads us to define an analogous of Nambooripad's *natural order* [16] on regular monoids.

2.1 Left and right shifts

For all tile non zero $u \in T_A$ we define the *left shift* $\sigma_L(u)$ of tile u to be the tile $\sigma_L(u) = (u_1u_2, 1, u_3)$ when $u = (u_1, u_2, u_3)$, i.e. $\sigma_L(u)$ is the tile obtained from u by shifting the root of u to the left.



Symmetrically, for all non zero tile $u \in T_A$ we define the *right shift* $\sigma_R(u)$ of tile u to be the tile 0 when $u = 0$ or the tile $\sigma_R(u) = (u_1, 1, u_2u_3)$ when $u = (u_1, u_2, u_3)$, i.e. $\sigma_R(u)$ is the tile obtained from u by shifting the root of u to the right.



Left and right shifts operators are extended to 0 by taking $\sigma_L(0) = \sigma_R(0) = 0$.

For all word $u \in A^*$, let u_C be the tile defined by $u_C = (1, u, 1)$: tile u_C is called the *root image* of word u. Let also u_L be the tile $u_L = (u, 1, 1) = \sigma_L(u_C)$ and let u_R be the tile $u_R = (1, 1, u) = \sigma_R(u_C)$.

Theorem 2 *The monoid of positive tiles T_A is finitely generated from* 0, 1, *root images of letters of A, product and left and right shift operators.*

Proof. The mapping $u \mapsto u_C$ from A^* to T_A is one to one monoid morphism. It follows that root images of letters generate, by product, root images of arbitrary words. We conclude with the fact that, for all u, v and $w \in A^*$, $(u, v, w) = u_L v_C w_R = \sigma_L(u_C) v_C \sigma_R(w_C)$. $\square$

2.2 Context elements

Images of elements of T_A by left and right shifts are called *context tiles* or *context elements*. The set of context tiles is denoted by C_A. It is an easy observation that both left and right shift operators define projections from T_A into C_A, i.e. for all $u \in C_A$, $\sigma_L(u) = \sigma_R(u) = u$.

Context elements themselves have many more properties that are stated in the next lemma:

Lemma 3 *Context elements form a submonoid C_A of T_A. They are commuting idempotents, i.e. for all $u \in C_A$, $uu = u$ and for all u and $v \in C_A$, $uv = vu$. Moreover, for all $u \in T_A$, $\sigma_R(u)u = u\sigma_L(u) = u$, i.e. left and right shifts of u are right and left local units of u.*

Proof. This proof presents no difficulties. It is essentially a matter of understanding product of tiles. $\square$

2.3 Natural order

The following definition extends to the monoid of positive tiles Nambooripad's natural order in inverse or regular monoids [16].

We say that a tile (u_1, u_2, u_3) is smaller than a tile (v_1, v_2, v_3), which we write $(u_1, u_2, u_3) \leq (v_1, v_2, v_3)$, when $v_1 \leq_s u_1$, $v_2 = u_2$ and $v_3 \leq_p u_3$. We extend this relation with $0 \leq u$ for all tile $u \in T_A$. One can easily check that this relation is an order. Some of its main properties are described in the following three lemmas.

Lemma 4 (The submonoid of root images) *Maximal elements of T_A under the natural order are the root images of words of A^*.*

Proof. Root images are obviously maximal elements and, by definition of the product of tiles, the product of two root images is a root image. $\square$

Lemma 5 (The meet semi-lattice of contexts) *For all context tiles u and $v \in C_A$, $u \leq v$ if and only if $uv = u$, and, moreover, $u \wedge v = uv$.*

Proof. Elements of C_A are commuting idempotent elements hence it is a classical result that C_A is a meet semi-lattice with respect to the order relation defined by $x \preceq y$ when $xy = x$. In that case, $x \wedge y = xy$. Then we easily check that this order $\preceq$ and the natural order $\leq$ coincide. □

Lemma 6 (Natural order and shift operators) *For all tiles u and $v \in T_A$, $u \leq v$ if and only if $u = \sigma_R(u).v$ if and only if $u = v.\sigma_L(u)$.*

Proof. The proof arguments are essentially the same as with inverse monoid (see [10] p21, Lemma 6) with $\sigma_L(u)$ *behaving like* $u^{-1}u$ and $\sigma_R(u)$ *behaving like* uu^{-1}. □

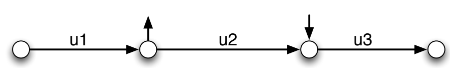
Remark. Such a notion of monoid, that behaves *like* inverse monoids without being truly inverse monoids, is further developed in [7] and [6].

3 Inverse completion

It occurs that the monoid T_A of positive tiles can be embedded in a larger (inverse) monoidM_A known as McAlister monoid [11]. This is achieved via completing the monoid of positive tiles by negative tiles.

3.1 Negative tiles and generalized product

A *negative (or right to left) tile* over the alphabet A is define to be a triple of word of the form $v = (u_1u_2, \bar{u}_2, u_2u_3) \in A^* \times \bar{A}^* \times A^*$ with u_1, u_2 and $u_3 \in A^*$. Such a tile is conveniently drawn as a Munn's birooted word tree



The set of positive and negative tiles extended with 0 is denoted by M_A.

For any two non zero tiles $u = (u_1, u_2, u_3)$ and $v = (v_1, v_2, v_3) \in M_A$, the sequential product of u and v is defined by

$$uv = ((u_1u_2 \vee_s v_1).\bar{u}_2, u_2v_2, \bar{v}_2(u_3 \vee_s v_2v_3))$$

when both $u_1u_2 \vee_s v_1 \neq 0$ and $u_3 \vee_s v_2v_3 \neq 0$, and $uv = 0$ otherwise. In this definition, word (and inverse word) products are now defined *in the free group* $FG(A)$ where, in particular, for all $u \in A^*$, $u\bar{u} = \bar{u}u = 1$

Theorem 7 *The set M_A equipped with the generalized sequential product is a monoid with neutral element 1 and absorbing element 0. The set $T_A \subseteq M_A$ of positive tiles is a submonoid of M_A.*

Proof. Observe first that the generalized product is well defined. For all u and v as above, we indeed have $(u_1u_2 \vee_s v_1).\bar{u}_2 \in A^*$, $\bar{v}_2(u_3 \vee_s v_2v_3) \in A^*$. Moreover, when $u_2v_2 \in \bar{A}^*$ (with elements of the free group $FG(A)$ always reduced), we also have $\overline{u_2v_2} = \bar{v}_2.\bar{u}_2 \leq_s (u_1u_2 \vee_s v_1).\bar{u}_2$ and $\overline{u_2v_2} = \bar{v}_2.\bar{u}_2 \leq_p \bar{v}_2(u_3 \vee_s v_2v_3)$.

Obviously, $1 = (1, 1, 1)$ is a neutral element, and, as for positive tiles, associativity follows from the fact that the domain of the product uv contains the domain of u and the domain of v.

Monoid T_A is a submonoid of monoid M_A since, when both u and v are positive tiles, the definition of the product in M_A just equals the definition of the product in T_A since, for all x and $y \in A^*$, when non zero, $x^{-1}(y) = \bar{x}y$ and $(x)y^{-1} = x\bar{y}$. □

Left and right shift operators are extended to all elements of M_A : for all $u = (u_1, u_2, u_3)$, $\sigma_L(u) = (u_1u_2, 1, u_3)$ and $\sigma_R(u) = (u_1, 1, u_2, u_3)$. Observe that non zero context elements, of the form $(u_1, 1, u_3)$, are exactly those elements that are both positive and negative tiles.

3.2 Inverses and shifts

In the picture above of the negative tile $v = (u_1u_2, \bar{u}_2, u_2u_3)$, it seems that tile v is obtained from tile $u = (u_1, u_2, u_3)$ just by swapping the input and output arrows. This observation justifies the following definition.

For all non-zero tile $u = (u_1, u_3, u_3) \in M_A$, let $u^{-1} \in M_A$ be the tile u^{-1}, the *(pseudo) inverse* of tile u, defined by $u^{-1} = (u_1u_2, \bar{u}_2, u_2u_3)$ and let $0^{-1} = 0$.

Lemma 8 *For all $u \in M_A$, $(u^{-1})^{-1} = u$, $\sigma_R(u) = uu^{-1}$, $\sigma_L(u) = u^{-1}u$ and $uu^{-1}u = u$.*

Proof. The case of 0 is obvious. Let $u = (u_1, u_2, u_3)$ and $u^{-1} = (u_1u_2, \bar{u_2}, u_2u_3)$. By definition of the (generalized) product, $uu^{-1} = ((u_1u_2 \vee_s u_1u_2)\bar{u}_2, u_2\bar{u_2}, u_2(u_3 \vee_p \bar{u_2}u_2u_3))$ hence, after simplification, $uu^{-1} = (u_1, 1, u_2u_3) = \sigma_R(u)$. Now, by definition again, $u\sigma_R(u) = (u_1, u_2.1, \bar{u}_2(u_2u_3 \vee_p u_2u_3))$ hence $u\sigma_R(u) = u$. By symmetrical arguments, we prove that $u^{-1}u = \sigma_L(u)$. The last statement follows from these facts. □

In other words, left and right shifts, that may look like adhoc operators over positive tiles, find a rather strong justification within the monoid of arbitrary tiles.

Theorem 9 *Monoid M_A is an inverse monoid.*

Proof. By Lemma 8, any tile $u \in M_A$ has a pseudo inverse u^{-1} hence M_A is regular. Commutation idempotent elements commute and this follows from Lemma 5 since idempotents of M_A are just context tiles hence M_A is inverse. □

3.3 McAlister monoid

McAlister monoid [14] is the quotient of the free inverse monoid $FIM(A)$ by the ideal $\perp$ of non unidirectional and non linear tiles.

More precisely, following Munn's result [15], elements of $FIM(A)$ are seen as birooted word trees, i.e. pairs (P, u) where P is a non empty finite and prefix-closed subset of (reduced elements of) the free group $FG(A)$ generated by A, with $u \in P$. The product of two birooted trees (P, u) and (Q, v) is defined by $(P, u).(Q, v) = (P \cup uQ, uv)$.

A birooted word tree (P, u) is said *unidirectional* when $P \subseteq A^* + \bar{A}^*$, and *linear* when both $P \cap A^*$ and $P \cap \bar{A}^*$ are totally ordered by the prefix order. It is straightforward that the set $\perp$ of non-unidirectional or non-linear birooted trees is an ideal.

The monoid of McAlister is then defined as the Rees quotient $FIM(A)/\perp$. In that monoid, given two linear and unidirectional birooted word trees (P, u) and (Q, v), the product of these two tiles is defined to be $(P \cup uQ, uv)$ as in $FIM(A)$ when the resulting birooted tree is linear and unidirectional, and $\perp$ (from now on written 0) otherwise.

Theorem 10 *The monoid M_A is isomorphic to the monoid of McAlister.*

Proof. (sketch of) For all non zero tile $u = (u_1, u_2, u_3) \in M_A$ let $t_u = (P_u, u_2)$ be the resulting birooted tree defined by $P_u = \{x \in \bar{A}^* : x \leq_p \bar{u}_1\} \cup \{x \in A^* : x \leq_p u_2u_3\}$.

We observe that t_u is a well-defined unidirectional and linear birooted tree. Indeed, when u is a positive tile, we have $u_2 \leq u_2u_3$ hence $u_2 \in P_u$. When u is a negative tile, i.e. with $u_2 \in \bar{A}^*$ we have both u_1u_2 and $u_2u_3 \in A^*$ and thus $u_2 \leq_p \bar{u}_1$ hence $u_2 \in P_u$.

We conclude then by showing that the mapping $\varphi : M_A \to FIM(A)/\perp$ defined by $\varphi(0) = 0$ and for any non-zero tile $u = (u_1, u_2, u_3) \in M_A$, $\varphi(u) = (P_u, u_2)$ is an isomorphism.

First, it is easy to check that it is a bijection. In fact, given a linear and unidirectional tile (P, u) one check that $\varphi^{-1}((P, u))$ is the tile (u_1, u_2, u_3) defined by $u_2 = u$, $u_1 = \bigvee_s \overline{P \cap \bar{A}^*}$ and $u_3 = \bar{u}_2 \bigvee_p P \cap A^*$.

It remains to show that it is indeed monoid morphism. For this, it is enough to check that for any two non zero tiles $u = (u_1, u_2, u_3)$ and $v = (v_1, v_2, v_3) \in M_A$, one indeed has $t_{u.v} = (P_u \cup u_2P_v, u_2v_2)$ which is straightforward. $\square$

Since McAlister is an inverse monoid, this gives another proof that M_A is inverse. For more details, we refer the interested reader to the book of Lawson on inverse monoids [10] where (yet another presentation of) McAlister monoid is given in chapter 9.4.

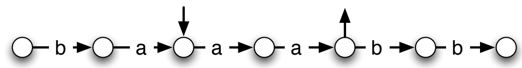
4 MSO-definable languages of tiles

We consider in this section the class MSO of languages of tiles definable by means of monadic second order formulae.

4.1 MSO definability

We need FO-models for positive tiles. For this, we use a typical encoding of words into FO-structures that amounts to encode each letter $a \in A$ as a relation between elements of the domain. This way, there is no need of end markers and the empty word is simply modeled by the structure with singleton domains and empty relations. We raise models of words to models of tiles just by marking (as for birooted trees) entry and exit points.

For instance, the triple $u = (ba, aa, bb)$ is modeled as indicated by the following picture



where, as before, a dangling input arrow marks the entry point and a dangling output arrow marks the exit point.

The model of a tile u is denoted by t_u. The associated domain of its underlying FO-structure is written $dom(t_u)$, the entry point in written $in(t_u)$ and the exit point is written $out(t_u)$.

A language $L \subseteq T_A$ is MSO definable when there is a MSO formula of the form $\varphi_L(U, x, y)$ where U is a set variable and x and y are two FO-variables such that, for all $t \in T_A$, $t \in L$ if and only if $t \models \varphi_L(dom(t), in(t), out(t))$.

4.2 A word congruence for languages of tiles

We aim at achieving a simple characterization of MSO definable language of positive tiles. For this purpose, we first define a notion of congruence relation over A^* which is defined for all language of tiles. It occurs that this congruence is of finite index if and only if the language of tile is definable in MSO.

Given a language $L \subseteq T_A - 0$, we define then the *word* congruence $\simeq_L$ associated to L as the least relation over words such that, for all u and $v \in A^*$, $u \simeq_L v$ when for all w_1, w_2 and $w_3 \in A^*$, the following equivalences hold: $(w_1uw_2, w_3, w_4) \in L \Leftrightarrow (w_1vw_2, w_3, w_4) \in L$, $(w_1, w_2uw_3, w_4) \in L \Leftrightarrow (w_1, w_2vw_3, w_4) \in L$, and $(w_1, w_2, w_3uw_4) \in L \Leftrightarrow (w_1, w_2, w_3vw_4) \in L$.

Theorem 11 *Let $L \subseteq T_A - 0$, L is MSO-definable if and only if relation $\simeq_L$ is of finite index.*

Proof. Let $L \subseteq T_A - 0$ be a language of tiles. The fact that this relation is a congruence is immediate from its definition. Now, $L = \Sigma_{(u,v,w) \in L}[u]_L \times [v]_L \times [w]_L$. Indeed, this means that whenever $(u, v, w) \in L$ for some u, v and $w \in A^*$ then if $u \simeq_L u'$, $v \simeq_L v'$ and $w \simeq_L w'$ for some u', v' and $w' \in A^*$, we also have $(u', v', w') \in L$. This property just follows from the definition of the word congruence $\simeq_L$.

Moreover, if $\simeq_L$ is of finite index, any language of the form $[w] \subseteq A^*$ with $w \in A^*$ is rational henceforth MSO definable, and the above sum is actually finite, henceforth there exists an MSO formula $\varphi(U, x, y)$ over tiles that defines L.

Conversely, we observe that language $L \subseteq T_A$ can be encoded into a language of words $M \subseteq A_P^* A_R^* A_S^*$ where A_P, A_R and A_S are three disjoint copies of the alphabet A for encoding left constraints, roots and right constraints of tiles.

If L is MSO definable then, clearly, so is M. Thus, by Büchi theorem, this means that M is also rational hence M is recognizable and thus its syntactic congruence $\simeq_M$ is of finite index hence so is $\simeq_L$. Indeed, for all word u and $v \in A^*$, we have $u \simeq_L v$ if and only if $u_X \simeq_M v_X$ for X being P, R or S and with w_X denoting the renaming of any word $w \in A^*$ in the copy alphabet A_X. □

5 Rational languages of tiles

The class RAT_S of languages of positive tiles is defined as the class of languages definable by means of finite sets of positive tiles, sum, product, iterated product (or star) and left and right shifts.

5.1 Rational expressions for MSO definable languages

A reformulation of Theorem 11 shows that all MSO definable language of tiles is rational in the above sense. More precisely:

Theorem 12 *A language of non-zero positive tiles $L \subseteq T_A - 0$ is MSO definable if and only if there are finitely many rational langages P_i, R_i and $S_i \subseteq A^*$ with $i \in I$ such that $L = \Sigma_{i\in I}(P_i)_L(R_i)_C(S_i)_L$. In particular $MSO \subseteq RAT_S$.*

Proof. Let $L \subseteq T_A - 0$. By definition of $\simeq_L$, we have already seen that $L = \sum_{(u,v,w)\in L}[u]_L \times [v]_L \times [w]_L$. By Theorem 11, we know that $\simeq_L$ is of finite index if and only if L is MSO definable. It follows, that for all $u \in A^*$, language $L_u = \{1\} \times [u]_L \times \{1\}$ belongs to the class RAT_S since the submonoid of maximal elements of T_A is isomorphic to A^* (see Lemma 4). But then, the above equality can be restated as $L = \sum_{(u,v,w)\in L} \sigma_L(L_u)L_v\sigma_R(L_v)$ hence, since this sum is finite whenever L is MSO definable, L is in the class RAT_S. □

5.2 MSO definability of rational languages

We aim here at proving that $RAT_S \subseteq MSO$. As finite sets of non-zero tiles are obviously definable in MSO, it suffices to prove adequate closure properties of the class of MSO definable languages of tiles.

Theorem 13 *For all L and $M \subseteq T_A - 0$, if L and M are MSO definable then so is $L + M$, $L.M$, L^*, $\sigma_L(M)$ and $\sigma_R(M)$. In particular, $RAT_S \subseteq MSO$.*

Proof. Let $\varphi_L(U, x, y)$ and $\varphi_M(U, x, y)$ be two formulae defining respectively the language of tiles L and M. We assume that these formulae also check that both x and y belongs to U with x before y (following the order induced by the relations) and that U is connected.

Case of $L + M$: take $\psi(U, x, y) = \varphi_L(U, x, y) \vee \varphi_M(U, x, y)$.

Case of $L.M$: take $\psi(U, x, y)$ stating that there exist two sets X and Y such that $U = X \cup Y$ and there is z such that both $\varphi_L(X, x, z)$ and $\varphi_M(Y, z, y)$ hold.

Case of L^* : in order to define $\varphi(U,x,y)$, the main idea is to consider the reflexive transitive closure $R^+(x,y)$ of the binary relation $R(x_1,x_2)$ defined by $\exists X \varphi_L(X,x_1,x_2)$; one must take care, however, that set U is completely covered by (sub)tiles' domains; this is equivalent to the fact, as domains necessarily overlap, that each extremity (left most or right most element) of the domain U belongs to one of these sets X at least. This is easily encoded by a disjunction of the three possible cases: extremities are reached in a single intermediate tile, left extremity is reached first or right extremity is reached first.

Case of $\sigma_R(M)$ (resp. $\sigma_L(M)$) : take $\psi(U,x,y)$ stating that $x=y$ and there is some z such that $\varphi_M(U,x,z)$ (resp. $\varphi_M(U,z,y)$). $\square$

With both Theorem 13 and Theorem 12 we have:

Corollary 14 $RAT_S = MSO$

Remark. All these results extend to languages of tiles possibly containing the undefined tile 0 by any adhoc definable encoding of tile 0 into a structure distinct from all other encodings of non zero tiles.

Remark. As in classical language theory, let left and right residuals of languages of tiles, i.e. $U^{-1}(W) = \{v \in T_A : \exists u \in U, uv \in W\}$ and $(W)V^{-1} = \{u \in T_A : \exists v \in V, uv \in W\}$. It can be shown that these operators are still definable in MSO. However, characterizing the expressive power of the resulting class of rational languages of tiles RAT_R, with residuals instead of shifts, remains an intriguing open problem.

6 Recognizable languages of tiles

In this section, we consider languages of arbitrary or positives tiles that are recognizable in the algebraic sense. Although the theory of tiles can be seen as part of the theory of inverse monoid, the results we obtain rather differ from the former studies of languages of words recognized by finite inverse monoids [13] or free inverse monoid languages [18]. Morphisms from T_A to arbitrary monoids turns out to be even more constraints than morphisms from A^* to finite inverse monoids.

6.1 Safely recognizable languages

Following the standard definition, we say that a language $L \subseteq T_A$ (resp. $L \subseteq M_A$) is a recognizable language of positive tiles (resp. arbitrary tiles) when there is a (surjective) monoid morphism $\varphi : T_A \to S$ (resp. $\varphi : M_A \to S$), with S a finite monoid such that $L = \varphi^{-1}(\varphi(L))$.

The syntactic congruence $\simeq_L$ over A^* associated to language L is defined, for all u and $v \in T_A$ (resp. u and $v \in M_A$), by $u \simeq_L v$ when for all x and $y \in T_A$, $xuy \in L \Leftrightarrow xvy \in L$. As well known, language L is recognizable if and only if its syntactic congruence is of finite index.

But this is not enough for developing a language theory of positive tiles. Indeed, the following lemma tells us that for achieving an adequate notion of recognizable languages of tiles, we need to restrict this definition.

Lemma 15 *Let $w \in A^\omega$ be an arbitrary infinite word on the alphabet A. The language of positive tiles $L_w = \{(1,1,v) \in T_A : 1 <_p v <_p w\}$ is recognizable. In particular, even if w is not computable, so is L_w while it remains recognizable.*

Proof. Let $S = \{0, 1, s\}$ be the monoid defined by usual product rules for 0 and 1 and $ss = s$, and let $\varphi_w : T_A \to S$ be the mapping defined for all $q \in T_A$ by $\varphi_w(q) = 1$ when $q = 1$, $\varphi_w(q) = s$ when $q \in L_w$ and $\varphi_w(q) = 0$ in the remaining cases.

We claim that φ_w is a morphism monoid. Indeed, for all q_1 and $q_2 \in L_w$ we have $q_1q_2 = q_1$ or $q_1q_2 = q_2$ hence $\varphi(q_1q_2) = \varphi(q_1)\varphi(q_2)$. It follows that $L_w = \varphi_w^{-1}(\varphi_w(s))$. □

This undesirable property results from the fact that T_A is not finitely generated. It follows that there can be some element $u \in T_A$ such that $\varphi(\sigma_R(u)) \neq 0$ (or $\varphi(\sigma_L(u)) \neq 0$) while $\varphi(u) = 0$. In McAlister monoid, finitely generated, this cannot happen since for all $u \in M_A$, $\sigma_L(u) = u^{-1}u$ and $\sigma_R(u) = uu^{-1}$.

This justifies the following definition. A (surjective) monoid morphism $\varphi : T_A \to S$ is *safe* when there are two mappings $s \mapsto \sigma_L(s)$ and $s \mapsto \sigma_R(s)$ from S in S such that, for all $u \in T_A$,$\varphi(\sigma_L(u)) = \sigma_L(\varphi(u))$ and $\varphi(\sigma_R(u)) = \sigma_R(\varphi(u))$. The class REC of recognizable languages of positive tiles is thus defined as the class of *safely recognizable* languages of tiles, i.e. languages recognizable by means of safe morphisms.

Theorem 16 *Safely recognizable languages of tiles are MSO definable, i.e. REC $\subseteq$ MSO.*

Proof. Let $\varphi : T_A \to S$ be a safe surjective morphism with S a finite monoid and let $X \subseteq S$. We want to prove that $L = \varphi^{-1}(X)$ is definable in MSO.

Let φ_L, φ_C and φ_R be the mappings from A^* to S defined, for all $u \in A^*$, by $\varphi_L(u) = \varphi(u_L)$, $\varphi_C(u) = \varphi(u_C)$ and $\varphi_R(u) = \varphi(u_R)$ (see Section 2.1 for the definition of u_C, u_L and u_R).

By morphism assumption, that for all $(u,v,w) \in T_A$, since $(u,v,w) = u_Lv_Cw_R$, we have $(u,v,w) \in \varphi^{-1}(s)$ if and only if $\varphi_L(u)\varphi_C(v)\varphi_R(w) = s$. Since S is finite (henceforth with a finite product table) it remains thus to show that these three values are definable in MSO.

For mapping φ_C this is easy as $\varphi_C : A^* \to S$ is just a monoid morphism. For mapping φ_L and φ_R there might be a problem as they are not monoid morphism. However, the safety assumption still ensures that $\varphi_L(u)$ is definable from $\varphi_C(u)$ and that $\varphi_R(w)$ is definable from $\varphi_C(w)$. As M is finite, images of words by φ_C are MSO definable and so are images of words by φ_L and φ_R. □

6.2 A non recognizable MSO definable language of tiles

The next result, negative, tells us that rather simple (MSO definable) languages of tiles are not recognizable.

Lemma 17 *Language $L = \{(ba^m, a^n, 1) \in T_A : m, n \in I\!N\}$ with a and b two distinct letters, is not recognizable.*

Proof. We prove that the syntactical congruence $\simeq_L$ associated to L in T_A (or in M_A) is of infinite index.

For all $m \in \mathbb{N}$, let u_m be the tile $u_m = (ba^m, 1, 1)$. It is an easy observation that for all m and $n \in \mathbb{N}$, $u_m \simeq_L u_n$ if and only if $m = n$ hence the claim. Indeed, for all $k \in \mathbb{N}$ let $v_k = (a^k, 1, 1)$. We have for any given $m \in \mathbb{N}$, $u_m v_k \in L$ if and only if $k \leq m$. Now if for some m and $n \in \mathbb{N}$ one has $u_m \simeq_L u_n$ then for all $k \in \mathbb{N}$, $u_m v_k \in L$ if and only if $u_n v_k \in L$. It follows that, for all k, $k \leq m$ if and only if $k \leq n$, hence $m = n$. □

Since language L defined above is obviously MSO, we have:

Corollary 18 *$RAT_S \neq MSO$*

6.3 A (non-trivial) recognizable language

Before studying in the next section (safe) recognizable languages in full generality, we provide in this section a non trivial example of such a language. It illustrates the main characteristic of all recognizable languages of tiles : a strong link with tiles's cover of periodic bi-infinite words.

Building such an example essentially amounts to provide a (surjective) safe monoid morphism from T_A (or even M_A) onto some non-trivial finite monoid M. Here, the main idea is to type tiles, by means of a safe monoid morphism, according to their capacity to cover the bi-infinite word ${}^{\omega}(ab)(ab)^{\omega}$ with a and b two distinct letters.

In order to do so, let $M = \{0, 1, (a, 1, b), (b, 1, a), (b, a, b), (a, b, a)\}$ with product $\odot$ defined as expected for 0 and 1 and defined according to the following product table:

$\odot$	$(a,1,b)$	$(b,1,a)$	(b,a,b)	(a,b,a)
$(a,1,b)$	$(a,1,b)$	0	0	(a,b,a)
$(b,1,a)$	0	$(b,1,a)$	(b,a,b)	0
(b,a,b)	(b,a,b)	0	0	$(b,1,a)$
(a,b,a)	0	(a,b,a)	$(a,1,b)$	0

Lemma 19 *Monoid $(M, \odot)$ is an inverse monoid.*

Proof. We easily check that product $\odot$ is associative hence M is a monoid. Given $E(M) = \{0, 1, (a, 1, b), (b, 1, a)\}$ the set of idempotents of S, the commutation of idempotents immediately follows from unique non trivial case $(a, 1, b) \odot (b, 1, a) = (b, 1, a) \odot (a, 1, b) = 0$. Last, we check that $(a, b, a) \odot (b, a, b) \odot$

$(a,b,a) = (a,b,a)$ and $(b,a,b) \odot (a,b,a) \odot (b,a,b) = (b,a,b)$. It follows that $(a,b,a)^{-1} = (b,a,b)$ and $(b,a,b)^{-1} = (a,b,a)$. All other element is idempotent and thus self-inverse. $\square$

The expected monoid morphism $\varphi : M_A \to M$ is then defined by $\varphi(0) = 0$, $\varphi(1) = 1$ and for all $(u,v,w) \in M_A$ such that $uvw \neq 1$, $\varphi(u,v,w) = 0$ when uvw is not a factor of $(ab)^\omega$ and, otherwise, when u is a positive tile:

1. $\varphi(u,v,w) = (a,1,b)$ when $|v|$ is even with $a \leq_s u$,$b \leq_p v$, $a \leq_s v$ or $b \leq_p w$,
2. $\varphi(u,v,w) = (b,1,a)$ when $|v|$ is even with $b \leq_s u$,$a \leq_p v$, $b \leq_s v$ or $a \leq_p w$,
3. $\varphi(u,v,w) = (b,a,b)$ when $|v|$ is odd with $a \leq_p v$,
4. $\varphi(u,v,w) = (a,b,a)$ when $|v|$ is odd with $b \leq_p v$,

and $\varphi(u,v,w) = (\varphi(uv,\bar{v},vw))^{-1}$ when (u,v,w) is a negative tile.

Theorem 20 *The mapping $\varphi : M_A \to M$ is a safe (surjective) morphism.*

Proof. This follows from the fact that, for all u and $v \in M_A$, $\varphi(u) \odot \varphi(v) = \varphi(uv) = \varphi(\varphi(u)\varphi(v))$. This morphism is safe with $\sigma_L((b,a,b)) = (a,1,b)$, $\sigma_L((a,b,a)) = (b,1,a)$, $\sigma_R((b,a,b)) = (b,1,a)$ and $\sigma_R((a,b,a)) = (a,1,b)$. $\square$

Given $L_S = (ab)^* + b(ab)^*$, given $L_C = (ab)^*$, given $L_P = (ab)^* + (ab)^*a$, this theorem says, in particular, that the non trivial tile language $L_S \times L_C \times L_P - 1$ is safely recognizable since it equals $\varphi^{-1}((b,1,a))$.

6.4 More on recognizable languages of tiles

We conclude our study by showing that recognizable languages of tiles are essentially generalization of the example described above: languages of REC are essentially definable out of finitely many periodic bi-infinite words.

Let $\varphi : T_A \to S$ be a safe monoid morphism with finite M. Since we can always restrict S to $\varphi(T_A)$ and $\varphi(0)$ is a zero in the submonoid $\varphi(T_A)$, we assume, without loss of generality, that $M = \varphi(T_a)$ with $\varphi(0) = 0$. Now, by complement, understanding the structure of languages of tiles recognizable by φ amounts to understand the structure of languages of the form $\varphi^{-1}(s)$ for all non-zero element $s \in S$ which is the purpose of the following theorem.

Theorem 21 *Let $s \in M$ be a non-zero element. Let $L_s = \varphi^{-1}(s)$. There is $x \in A^*$, $y \in A^+$ and $k \geq 0$ such that, either L_s is finite with $L_s \subseteq \{u \in T_A : v \leq u\}$ for some $v \in T_A - 0$ or L_s is a* co-finite subset *of one of the following set of non-zero tiles:*

1. *$S(^\omega(xy)) \times x \times P(w)$ for some finite $w <_p (yx)^\omega$,*
2. *$S(w) \times x \times P((yx)^\omega)$ for some finite $w <_s {}^\omega(xy)$,*
3. *$S(^\omega(xy)) \times x \times P((xy)^\omega)$,*

4. *or* $S({}^{\omega}(xy)) \times x(yx)^k(yx)^* \times P((yx)^{\omega})$,

with, for all $w \in A^* + {}^{\omega}A$, $S(w) = \{z \in A^* : z <_s w\}$, *i.e. the set of strict suffix of* w, *and for all* $w \in A^* + A^{\omega}$, $P(w) = \{z \in A^* : z <_p w\}$, *i.e. the set of strict prefix of* w.

In all cases, tiles of L_s *are compatible with the bi-infinite periodic word* ${}^{\omega}(xy)x(yx)^{\omega}$.

Proof. The rest of this section is dedicated to the proof of this theorem. In order to do so, we first prove several closure properties of L_s.

Lemma 22 *For all* $u = (u_1, u_2, u_3) \in L_s$ *and* $v = (v_1, v_2, v_3) \in L_s$, $(u_1 \vee_s v_1, u_2, u_3) \in L_s$ *and* $(u_1, u_2, u_3 \vee_p v_3) \in L_s$.

Proof. We know that $(v_1)_L v = v$ hence $(v_1)u \in L_s$ since $\varphi((v_1)_L v) = \varphi_L(v_1)\varphi(v) = \varphi_L(v_1)\varphi(u) = \varphi((v_1)_L u)$. Moreover, $0 \notin L_s$ hence $(v_1)_L u = (u_1 \vee_s v_1, u_2, u_3)$ with $u_1 \vee_S v_1 \neq 0$. Symmetrical arguments prove the prefix case. □

Lemma 23 *For all* $u = (u_1, u_2, u_3)$ *and* $v = (v_1, v_2, v_3) \in L_s$ *such that* $|u_2| \leq |v_2|$, *either* $u_2 = v_2$ *or there exists* $x \in A^*$, $y \in A^+$ *and* $k \geq 0$ *such that* $u_2 = x(yx)^k$, $v_2 = x(yx)^{k+1}$. *In that latter case, there is* $u' \in L_s$ *such that* $u'((xy)_C)^* \subseteq L_s$.

Proof. Let u and v as above. We have $(v_2)_R v(v_2)_L = v$ hence, by a similar argument as in Lemma 22, $(v_2)_R u(v_2)_L \in L_s$. Since $|u_2 \leq |v_2|$ this means that $u_2 \leq_p v_2$ and $u_2 \leq_s v_2$, i.e. *roots of elements of* L_s *are totally by both prefix and suffix*.

In the case $u_2 \neq v_2$ this means $v_2 = wu_2 = u_2w'$ for w and $w' \in A^+$. Let then $k \geq 0$ be the greatest integer such that $|w^k| < |u_2|$.

When $k = 0$, this means $w = u_2y$ for some $y \in A^*$ and we take $x = u_2$. Otherwise, by a simple inductive argument over k, this means $u_2 = w^kx$ for some $x \in A^*$ with $|x| < |w|$. In that latter case, we have $v_2 = w^{k+1}x = w^kxw'$. Since $|w'| = |w|$ it follows that $w' = yx$ for some $y \in A^+$ and thus $w = xy$. In all cases, $u_2 = x(yx)^k$ and $v_2 = x(yx)^{k+1}$.

By applying Lemma 22, with $v'_1 = u_1 \vee_s v_1$, $v'_3 = u_3 \vee_p v_3$ and $v' = (v'_1, v_2, v'_3)$, we have $v' \in L_s$. But we known that $(v_2v'_3)_L v' = v'$ hence we also have $(v_2v'_3)_L u \in L_s$. By applying product definition, this means that $u' = (v'_1, u_2, yxv'_3) \in L_s$ hence $v' = u'(yx)_C \in L_s$ hence, by an immediate pumping argument, $u'((yx)_C)^* \subseteq L_s$. □

Lemma 23 already proves the finite case of Theorem 21. We assume now L_s is infinite.

In the case a single root appears in elements of L_s, for all $u = (u_1, x, u_3) \in L_s$ we have $(u_1)_L u(u_3)_L = u$ hence, because $s \neq 0$, $\varphi_L(u_1) \neq 0$ and $\varphi_L(u_3) \neq 0$. By *safety assumption*, this means that $\varphi_C(u_1) \neq 0$ and $\varphi_C(u_3) \neq 0$. Since M is *finite*, while L_s is finite, this means that two distinct constraints have same images by φ_C. Applying Lemma 23 this implies all domains of tiles of L_s are factors of the same periodic bi-infinite word of the form ${}^{\omega}(xy)x(yx)^{\omega}$ for some x and $y \in A^*$ with $xy \neq 0$.

Depending on the case elements of L_s have infinitely many left constraints, right constraints or both left and right constraints, we conclude by showing that for all $v_1 \leq_s {}^{\omega}(xy)$ and $v_3 \leq_p (yx)^{\omega}$ there is $v \in L_s$ such that either $(v_1)_L v$, $v(v_3)_R$ or $(v_1)_L v(v_3)_R \in L_s$ hence, given a given fixed $u = (u_1, x, u_3) \in L_S$, either $(v_1)_L u$, $u(v_3)_R$ or $(v_1)_L u(v_3)_R \in L_s$. This proves the co-finiteness inclusion in case 1, 2 and 3 in Theorem 21.

In the case two distinct roots (hence infinitely many) appears in tiles of L_s. Applying Lemma 23 there is $x \in A^*$, $y \in A^+$ and $k \geq 0$ with $u = (u_1, u_2, xyu_3)$ such that $u((xy)_C)^* \subseteq L_s$. By choosing properly the initial u and v in Lemma 23, we may assume that $u_2 = x(yx)^k$ is the least root of elements of L_s and that $y(yx)^{k+1}$ is the second least. Then we claim that all roots of elements of L_s belongs to $x(yx)^k(yx)^*$.

Indeed, let then $(v_1, v_2, v_3) \in L_s$ and let $m \geq 0$ be the unique integer such that $v_2 = x(yx)^{m+k}x'$ with $x' <_p x$ hence $x = x'y'$ for some $y' \in A^+$. By an argument similar with the argument in the proof of Lemma 23, we can show that there is $u' = (v_1', x(yx)^{m+k}, x'v_3') \in L_s$ with $v' = (v_1', v_2, v_3') \in L_s$ henceforth with $v' = u'x'_C \in L_s$. But this also means that $ux'_C = (u_1, u_2x', y'yu_3) \in L_s$ which, by minimality of u_2xy as second least roots, forces x' to be equal to 1. This shows the inclusion stated in case 4 in Theorem 21.

Co-finiteness of the inclusion follows from the finiteness of M with arguments similar to arguments for case 3 above. □

7 Open perspective: from discrete to timed tiles

In application oriented perspectives, tiles can be seen as models of process behaviors distinguishing a notion of realization window (domains or tiles) from a notion of synchronization window (tiles' roots), the pattern matching constraints modeling communications. Observe that this potentially leads to a rather rich alternative with process algebras that could be studied as such.

Moreover, especially modeling music with tiles, as in [5], we need to model time. This can easily be done extending the alphabet letters with positive durations, defining a notion of *timed tiles*. What is the expressive power of the resulting notion of rational expressions extended with shifts ? What is the expressive power of the resulting notion of rational expressions extended with (see Section 5.2) residuals ?

Compared to Dima's proposal for an expressive language of timed regular expressions, timed tiles seem simpler than timed dominoes [4]. Moreover, preliminary studies for timed tiles also show that all pathological exemples described in [2] are easily encoded by means of tiles expressions with shifts or even residuals.

References

[1] Filipa Soares de Almeida. *Algebraic Aspects of Tiling Semigroups*. PhD thesis, Universidade de Lisboa, Faculdade de Ciências Departamento de Matemática, 2010.

[2] Eugene Asarin, Paul Caspi, and Oded Maler. Timed regular expressions. *J. ACM*, 49(2):172–206, 2002.

[3] Jean-Camille Birget. Concatenation of inputs in a two-way automation. *Theoretical Computer Science*, 63(2):141 – 156, 1989.

[4] Catalin Dima. Regular expressions with timed dominoes. In *DMTCS*, volume 2731 of *LNCS*, pages 141–154. DMTCS, 2003.

[5] David Janin. Modélisation compositionnelle des structures rythmiques : une exploration didactique. Technical Report RR-1455-11, LaBRI, Université de Bordeaux, 2011.

[6] David Janin. Quasi-inverse monoids. Technical Report RR-1459-12, LaBRI, Université de Bordeaux, 2012.

[7] David Janin. Quasi-recognizable vs MSO definable languages of one-dimentionnal overlaping tiles. Technical Report RR-1458-12, LaBRI, Université de Bordeaux, 2012.

[8] J. Kellendonk. The local structure of tilings and their integer group of coinvariants. *Comm. Math. Phys.*, 187:115–157, 1997.

[9] Michal Kunc and Alexander Okhotin. Describing periodicity in two-way deterministic finite automata using transformation semigroups. In *Developments in Language Theory*, volume 6795 of *Lecture Notes in Computer Science*, pages 324–336. Springer, 2011.

[10] Mark V. Lawson. *Inverse Semigroups : The theory of partial symmetries*. World Scientific, 1998.

[11] Mark V. Lawson. McAlister semigroups. *Journal of Algebra*, 202(1):276 – 294, 1998.

[12] Bertrand LeSaëc, Igor Litovsky, and Bruno Patrou. A more efficient notion of zigzag stability. *ITA*, 30(3):181–194, 1996.

[13] Stuart W. Margolis and Jean-Eric Pin. Languages and inverse semigroups. In *ICALP*, volume 172 of *Lecture Notes in Computer Science*, pages 337–346. Springer, 1984.

[14] D.B. McAlister. Inverse semigroups which are separated over a subsemigroups. *Trans. Amer. Math. Soc.*, 182:85–117, 1973.

[15] W. D. Munn. Free inverse semigroups. *Proceeedings of the London Mathematical Society*, 29(3):385–404, 1974.

[16] K. S. S. Nambooripad. The natural partial order on a regular semigroup. *Proc. Edinburgh Math. Soc.*, 23:249–260, 1980.

[17] Jean-Pierre Pécuchet. Automates boustrophedon, semi-groupe de birget et monoide inversif libre. *ITA*, 19(1):71–100, 1985.

[18] Pedro V. Silva. On free inverse monoid languages. *ITA*, 30(4):349–378, 1996.