Theoretical Computer Science 410 (2009) 4961–4974

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

On locally reversible languages☆

Pedro García, Manuel Vázquez de Parga, Antonio Cano, Damián López *

Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Camino de Vera s/n, 46071, Valencia, Spain

A R T I C L E I N F O

Article history:
Received 23 July 2007
Received in revised form 17 June 2009
Accepted 3 July 2009
Communicated by D. Perrin

Keywords:
Formal languages
Locally reversible language
Quasi-k-reversible automata

A B S T R A C T

There exist several works that study the class of reversible languages defined as the union closure of 0-reversible languages, their properties and suitable representations. In this work we define and study the class of locally reversible languages, defined as the union closure of k-reversible languages. We characterize the class and prove that it is a local (positive) variety of formal languages. We also extend the definition of quasi-reversible automata to deal with locally reversible languages and propose a polynomial algorithm to obtain, for any given locally k-reversible language, a quasi-k-reversible automaton.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Many works have paid attention to reversible languages and their properties, as well as proper ways to represent them. The first implicit reference to them is due to McNaughton. He proved that the loop complexity is equal to the cycle rank of the reduced state graph. The first explicit reference to reversible languages is due to Angluin [1], who defines a hierarchy for reversible languages for different values of a parameter k, where the whole hierarchy is contained in the class of regular languages, and called it the class of reversible languages. Angluin's proves that, for any fixed k, the class is identifiable from positive presentation in the limit.

Further work on these languages was mainly focused on the class of 0-reversible languages and the union closure of 0-reversible languages. This leads to referring to this class as reversible languages, and therefore, further work on the hierarchy defined by Angluin refers to it as the class of k-reversible languages.

0-reversible languages are accepted by bideterministic automata, that is, automata with one initial state and one final state and whose transition function is both deterministic and codeterministic. Reversible languages are accepted by reversible automata, that can be defined as deterministic and codeterministic automata that allow multiple initial and final states.

Group languages are accepted by automata whose transition function defines a total one-to-one map from the set of states into itself. Therefore, reversible languages are a natural generalization of group languages because reversible automata define a partial one-to-one map on the set of states of the automaton. Furthermore, for any language L, it is a group language if and only if both L and its complementary are reversible languages. Several other works also consider reversible languages from different points of view, for instance: quantum finite automata [9], artificial intelligence [7], biprefix codes [10], inverse monoids [17], topological problems [11,13], etc.

The class of reversible languages, the union closure of 0-reversible languages, was studied by Pin in [13], where he gives several characterizations of the class. In [14], Pin gives an algorithm to decide whether or not a language belongs to the

☆ This work was partially supported by the Spanish Ministerio de Ciencia e Innovación (contracts TIN2007-60769 and HF2008-0009), and Generalitat Valenciana (contract GV06/068).

* Corresponding author. Tel.: +34 96 3877007; fax: +34 963877359.
E-mail addresses: pgarcia@dsic.upv.es (P. García), mvazquez@dsic.upv.es (M. Vázquez de Parga), acano@dsic.upv.es (A. Cano), dlopez@dsic.upv.es (D. López).

0304-3975/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2009.07.009

class and he proves that, for some reversible languages, the minimal deterministic finite automaton (DFA) is exponentially bigger than a reversible automaton for the same language. In a related work, Héam [11] finds that there exists another family of reversible languages such that, for any given language of the family, the minimal reversible automaton is exponentially bigger than the minimal DFA for the same language. These results imply that both representations are incomparable.

The most suitable representation for reversible languages is due to Lombardy [12], who proves that every reversible language can be represented by a *quasi-reversible automaton*. This family of automata include reversible ones, have similar properties, and furthermore, any quasi-reversible automaton can be easily modified to obtain a reversible one. Lombardy gives a bound of 2^n for the size of the quasi-reversible automaton, where n denotes the size of the minimal DFA. Recently, there has been proposed a polynomial algorithm to obtain a quasi-reversible automata for any given reversible language [6]. In that work, a new bound of n for the size of the automaton is also given.

In the vein of Pin's definition of the class of reversible languages, in this work we study *locally k-reversible languages*, defined as the union closure of Angluin's k-reversible languages. We prove that this class is a positive variety of formal languages. In order to represent these languages in a proper way, we extend the definition of quasi-reversible automata and define the *quasi-k-reversible* automata. We also propose an algorithm to obtain, from the minimal DFA of a locally k-reversible language, a quasi-k-reversible automaton. This automaton can be easily processed to obtain a locally k-reversible one. It is expected that locally k-reversible languages will be as useful in those fields in which reversible languages are.

Among others, an important formalism we use is the saturated *residual finite state automaton* (RFSA), introduced and developed by Denis, Lemay and Terlutte [2–4]. The size of the quasi-k-reversible automaton we obtain is bounded by the number of states of the minimal DFA and can be easily modified to obtain a collection of locally k-reversible automata that accepts the language.

2. Definitions and notation

Let Σ be an alphabet, let Σ^* denote the free monoid of words or strings over Σ, and let ϵ denote the empty word. Given a string u over the alphabet we denote its length by $|u|$. A language over Σ is any set $L \subseteq \Sigma^*$. For any given word u and language L, let $u^{-1}L$ denote the derivative of L by u.

A finite automaton is defined as a tuple $A = (Q, \Sigma, \delta, I, F)$, where Q is a finite set of states, Σ is an alphabet, $I, F \subseteq Q$ are respectively the set of initial and final states, and δ is the transition function defined from $Q \times \Sigma$ into $\mathcal{P}(Q)$. This function can be extended in order to consider words over an alphabet instead of symbols as well as a set of states instead of a single one. For the sake of clarity, we will usually consider the transition function as a subset of $Q \times \Sigma \times Q$. The language accepted by the automaton is $L(A) = \{u \in \Sigma^* : \delta(I, u) \cap F \neq \emptyset\}$. Given two states p, q of an automaton, we say that there is a *path* between p and q whenever there is a word u such that $q \in \delta(p, u)$. An automaton is *trimmed* if, for each state p, there are words u, v such that p is in $\delta(I, u)$ and $\delta(p, v)$ contains a final state. In this work we will consider that all the automata have been trimmed. A finite automaton is deterministic (i.e. is a DFA) if it has only one initial state and the transition function is defined in such a way that, for any state q and any symbol a, $\delta(p, a)$ has at most one element.

A finite automaton is said to be k-reversible if it is deterministic and fulfills the following conditions: (i) whenever there are transitions (p_1, a, r) and (p_2, a, r) in the automaton there is no word u of length k reaching both p_1 and p_2. and (ii) there is no word u of length k that reaches two distinct final states. A finite automaton is said to be reversible if each symbol induces a partial one-to-one map from the set of states into itself. A language is said to be k-reversible (reversible) if it is accepted by a k-reversible (reversible) automaton.

Given a reversible language L, the minimal DFA for L is usually not reversible. Note that the class of reversible languages results from the union closure of 0-reversible languages.

A *strongly connected component* (SCC) of an automaton is a maximal subautomaton such that there is a path between every pair of states.

Given an automaton A and a state q of the automaton, let the *right language* of the state q be defined as $R_q^A = \{x \in \Sigma^* : \delta(q, x) \cap F \neq \emptyset\}$. Given any two states p, q of the automaton, let $<$ be the relation defined as $p < q$ if and only if $R_p^A \subseteq R_q^A$.

Let us define the set $D = \{u^{-1}L : u \in \Sigma^*\}$ and the set $U = \{u_1^{-1}L \cap \cdots \cap u_k^{-1}L : k \geq 0, u_1, \ldots, u_k \in \Sigma^*\}$. Note that whenever the language L is regular, the sets D and U are finite.

The minimal DFA for a language L is $A = (D, \Sigma, \delta, \{q_0\}, F)$, where $q_0 = \epsilon^{-1}L = L$; $F = \{u^{-1}L : u \in L\}$; and $\delta(u^{-1}L, a) = (ua)^{-1}L$ for any state in D and any symbol a of the alphabet. The *universal automaton* for a language L is $\mathcal{U} = (U, \Sigma, \delta, I, F)$, where $I = \{p \in U : p \subseteq L\}$, $F = \{p \in U : \epsilon \in p\}$ and the set of transitions is defined as $\delta(u^{-1}L, a) = \{p \in U : p \subseteq (ua)^{-1}L\}$.

Given a language L, a *residual finite state automaton* (RFSA) for the language is a finite automaton A that accepts the language L, and for any state q, there exists a word u such that $R_q^A = u^{-1}L$. The *saturated* RFSA of a minimal DFA A is defined as the subautomaton of $\mathcal{U}$ induced by the set of states of the automaton A.

Let us define the function $f_k : \Sigma^{\geq k} \to \Sigma^k$, where $f_k(x)$ is the suffix of length k of the word x.

A *semigroup* is formed by a set S and an associative internal operation, usually denoted in a multiplicative way. A *monoid* is defined as a semigroup that has a neutral element denoted by 1. An idempotent of the monoid is an element e such that $ee = e$. The set of idempotents of a semigroup S (a monoid M) is denoted by $E(S)$ ($E(M)$). Given an element of the monoid

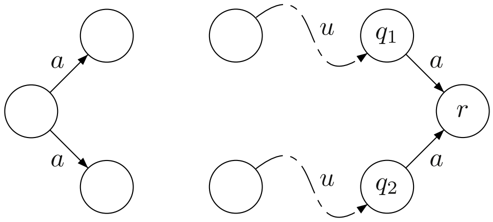


Fig. 1. Configurations that cannot appear in locally k-reversible automata, where u is a word of length k.

x, we denote by x^ω the unique idempotent that we can obtain by multiplying x with itself. Given a semigroup S, the local monoid of S associated to a idempotent e of S is defined as eSe.

Given a monoid M, we say that an order relation $\leq$ is stable on M if and only if, for any elements x, y, z of the monoid, $x \leq y$ implies that $xz \leq yz$ and $zx \leq zy$. A monoid M is said to be *ordered* if it is equipped with an stable order. Ordered monoids are usually denoted by $(M, \leq)$. Given an ordered monoid $(M, \leq)$, we say that $I \subseteq M$ is an *order ideal* if, for any x in the monoid such that $x \leq y$, where y is in I, then x is also in I.

Given a rational (regular) language L, the syntactic congruence over words $\sim_L$ is defined as $x \sim_L y$ if uxv in L if and only if uyv is also in L, where u, v are words over Σ^*. In the same way, the *syntactic quasi-preorder* $\preceq_L$ is defined as $x \preceq_L y$ if whenever the word uyv is in L then the word uxv is also in L.

The *syntactic monoid* of a language L is denoted and defined by $Syn(L) = \Sigma^*/\sim_L$. The previously defined syntactic quasi-preorder $\preceq_L$ induces a stable order over $Syn(L)$. In what follows, we will refer to this order as the *syntactic order* of L. The syntactic ordered monoid associated to L will be denoted by $(Syn(L), \preceq_L)$.

A variety of finite (ordered) monoids, or pseudovariety, is a class of finite monoids closed under submonoids, quotients, and finite direct products. A positive variety of languages is a class of recognizable languages $\mathcal{V}$ such that:

- for every alphabet Σ, $\mathcal{V}(\Sigma^*)$ is a positive Boolean algebra (closed under union and intersection),
- $\varphi\colon \Sigma^* \to \Gamma^*$ is a morphism of monoids, where $L \in \mathcal{V}(\Gamma^*)$ implies that $\varphi^{-1}(L) \in \mathcal{V}(\Sigma^*)$,
- if $L \in \mathcal{V}(\Sigma^*)$ and if $a \in \Sigma$, then $a^{-1}L$ and La^{-1} are in $\mathcal{V}(\Sigma^*)$.

A variety of languages is a positive variety closed under complementation.

Given two positive varieties of languages $\mathcal{V}$ and $\mathcal{W}$, we write $\mathcal{V} \subseteq \mathcal{W}$ if, for each alphabet Σ, $\mathcal{V}(\Sigma^*) \subseteq \mathcal{W}(\Sigma^*)$.

There is a one-to-one correspondence between varieties of finite monoids (resp. varieties of finite ordered monoids) and varieties of recognizable languages (resp. positive varieties of recognizable languages) [5,15].

A variety that will be of interest in this work is the variety of reversible languages, studied by Pin [13,14] and denoted E^-_{com}. This is the variety of ordered monoids where the idempotents commute and $1 \leq e$, for any idempotent e. *Ordered semigroups* can be defined in the same way that ordered monoids are.

Given a variety of ordered monoids V, a variety of semigroups is said to be locally in V, and denoted by LV, if every local monoid belongs to V. For further semigroup theory definitions and/or notation, we refer the reader to [16].

3. Locally *k*-reversible automata and languages

First, let us define the class of *locally k-reversible* languages and relate it to the previously defined classes of k-reversible and reversible languages.

Definition 3.1. Given an automaton $A = (Q, \Sigma, \delta, I, F)$, we say it is *locally k-reversible* if and only if the following conditions are fulfilled:

- the transition function is deterministic;
- for any given word u of length k, any states q_1, q_2, r, where $q_1 \neq q_2$, and any symbol a, if u reaches q_1 and q_2, then the automaton does not contain transitions (q_1, a, r) and (q_2, a, r).

Fig. 1 illustrates those forbidden configurations for locally k-reversible automata. A language is said to be *locally k-reversible* if there exists a locally k-reversible automaton that recognizes it. Note that locally k-reversible languages result from the closure union of k-reversible languages (multiple initial states are allowed and there can be a word of length k reaching two distinct final states). An example of locally k-reversible automaton is shown in Fig. 2. Note that for any locally k-reversible language L, its minimal DFA is usually not locally k-reversible; for instance, Fig. 3 shows the minimal DFA for the automaton in Fig. 2.

An anonymous referee stressed the unsymmetric definition of locally k-reversible automata with respect to reversible automata. He also suggest a more symmetric definition. We prove in the Appendix that both definitions are equivalent.

Locally k-reversible languages are related to k-reversible languages in the same way that reversible languages are related to 0-reversible languages; that is, the class of locally k-reversible languages is the union closure of k-reversible languages. In order to denote these classes of languages, let $kRev$ denote the class of k-reversible languages, let $LkRev$ denote the class of locally k-reversible languages and let $LRev$ denote the class of locally reversible languages, i.e. locally k-reversible for some value of k. Also, let Rev denote the class of reversible languages.

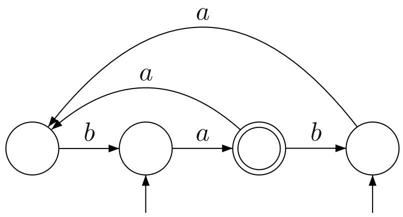


Fig. 2. Example of a locally 1-reversible automaton.

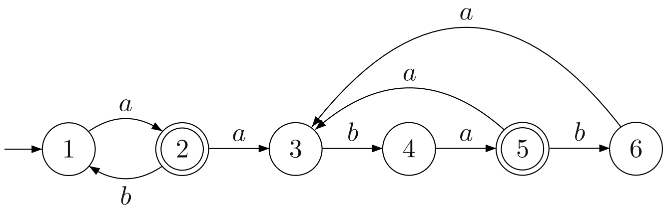


Fig. 3. Minimal DFA equivalent to the automaton in Fig. 2. This automaton is not locally 1-reversible because (i) there are transitions from states 2 and 5 to the state 3 with the same symbol and (ii) there is a word of length 1 that reaches both states 2 and 5.

4. Characterization of locally reversible languages

Pin proves in [13] that the class of languages Rev is a positive variety of languages (it is not closed under complementation). In the same paper, Pin proves that the ordered syntactic monoid of the languages in Rev is in $E^-_{com} = [[x^\omega y^\omega = y^\omega x^\omega,\ 1 \le x^\omega]]$. In this section we prove that the class $LRev$ is also a positive variety of languages that corresponds to LE^-_{com}. This is the reason why we refer to it as the class of locally reversible languages.

Let now consider the transducer $\tau_k = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$, where Q is the set $\bigcup_{i=0}^{k-1} \Sigma^i$; $q_0 = \epsilon$; Δ is the set Σ^k (we will enclose the symbols of Δ in square brackets in order to distinguish between the symbols of both alphabets); the transition function and the output one be defined as follows:

$$\delta(p, a) = \begin{cases} pa, & \text{if } |p| < k-1 \\ f_{k-1}(pa), & \text{otherwise;} \end{cases} \qquad \lambda(p, a) = \begin{cases} \epsilon, & \text{if } |p| < k-1 \\ [pa], & \text{otherwise} \end{cases}$$

for any given state p and symbol a of the alphabet.

Intuitively, the transducer τ_k outputs for each word u the sequence of segments of length k (for instance, $\tau_2(abba) = [ab][bb][ba]$). Given a language L, we will denote with $\tau_k(L)$ the transduction of L by the transducer τ_k. Fig. 4 shows the transducers τ_2 and τ_3. Let us denote the cascade product [8] of an automaton A and a transducer τ as $\tau \circ A$. Note that, for any automaton A, $\tau^{-1}(L(A))$ is equal to $L(\tau \circ A)$.

We now prove partial results that allow us to conclude that the class of locally reversible languages corresponds to the variety $E^-_{com} * LI$ (where $*$ denotes the composition operator).

Proposition 4.1. *Let L be a reversible language and let τ_{k+1} be the transducer defined above. Then $\tau^{-1}_{k+1}(L)$ is a locally k-reversible language.*

Proof. Let $B = (Q_B, \Delta, \delta_B, I, F)$ be a reversible automaton for the language. The automaton B is reversible; therefore it can be decomposed into a collection $\{B_i\}_{i=1}^n$ of bideterministic automata. In order to prove the proposition, we will prove that, for each $i : 1..n$, the automaton $\tau_{k+1} \circ B_i$ is a k-reversible automaton.

Let $B_i = (Q_i, \Delta, \delta_i, q_{oi}, q_{fi})$ be one automaton from the collection. The cascade product $\tau_{k+1} \circ B_i$ returns an automaton $A_i = (Q_\tau \times Q_i, \Sigma, \delta, (\epsilon, q_{oi}), Q_\tau \times q_{fi})$, where the transition function is defined as follows:

$$\delta((p, q), a) = (\delta_\tau(p, a), \delta_i(q, \lambda(p, a))$$

Note that if any of the automata A_i were not k-reversible there would be a word x (of length k) reaching the states (x, p_1) and (x, p_2) and also transitions $((x, p_1), b, (f_k(xb), p))$ and $((x, p_2), b, (f_k(xb), p))$. The existence of these transitions would imply that transitions of the form $(p_1, [xb], p)$ and $(p_2, [xb], p)$ would appear in B_i. This is impossible because B_i is bideterministic. Fig. 5 summarizes the situation. □

Proposition 4.2. *If L is a locally k-reversible language, then $\tau_{k+1}(L)$ is reversible.*

Proof. Let $\tau_{k+1} = (Q_\tau, \Sigma, \Delta, \delta_\tau, \lambda, q_0)$ be a transducer as stated above and let $A = (Q_A, \Sigma, \delta_A, I, F)$ be a locally k-reversible automaton. Also, let $Q'_\tau = Q_\tau - \bigcup_{i=0}^{k-1} \Sigma^i$ be a subset of states from the set of states of the transducer.

Let us define the automata $B = (Q'_\tau \times Q_A, \Delta, \delta', I', Q'_\tau \times F)$, where

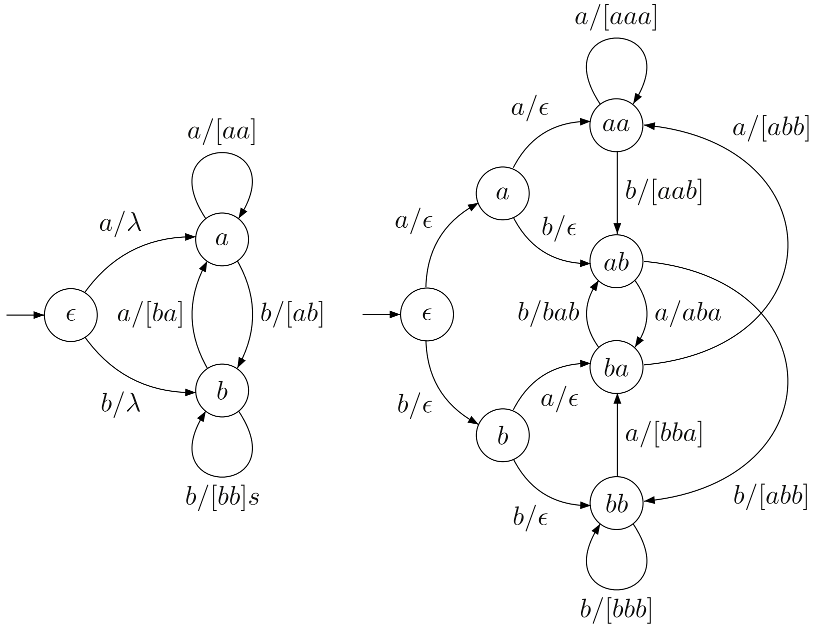


Fig. 4. Transducers τ_2 and τ_3.

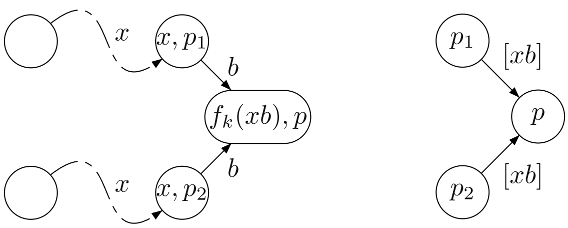


Fig. 5. The figure on the left shows the configuration of an automaton A that does not fulfill the k-reversibility conditions. If this automaton is obtained by the cascade product $\tau_{k+1} \circ B_i$, then the automaton B_i should contain the configuration on the right, which would make B_i not to be bideterministic.

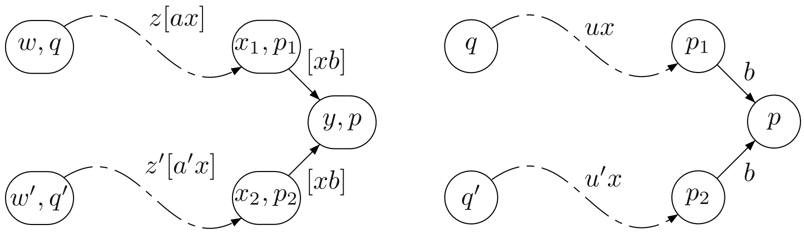


Fig. 6. The scheme on the left shows the configuration of a non-reversible automaton B. The scheme on the right shows the corresponding automaton A for the situation. Note that this automaton should be locally k-reversible and it is not.

- $I' = \{(x, q) \mid x \in Q_\tau^{'} \wedge q \in \delta_A(I, x)\}$;
- $\delta'((x, q), b) = (f_k(xa), \delta_A(q, a))$ for b in Δ, $x \in Q_\tau^{'}$ and $q \in Q_A$ and where $\lambda(x, a) = [xa] = b$. Note that the transition function δ' is well defined because both the transition and output functions of the transductor τ_{k+1} are deterministic.

It is easy to see that $L(B) = \tau_{k+1}(L \cap \Sigma^{k+1}\Sigma^*)$; that is, when the words of length lower than or equal to k are not considered, $L(B)$ is the transduction of the words in $L(A)$ into sequences of segments of length $k + 1$.

To prove that $\tau_{k+1}(L(A))$ is reversible it is enough to prove that the automaton B is reversible. If, by contradiction, it were not reversible, then there would be states (x_1, p_1), (x_2, p_2) and (y, p) in B and a symbol $[xb]$ in Δ (i.e. x in Σ^k and b in Σ) such that $((x1, p1), [xb], (y, p))$ and $((x1, p1), [xb], (y, p))$ would be transitions in δ'.

By construction, this situation would be possible only if x_1 is equal to x_2 and to x. In this situation, in the automaton A, there would be words ux and $u'x$ reaching respectively the states p_1 and p_2, together with transitions (p_1, b, p), and (p_2, b, p). Therefore A would not be locally k-reversible, which is a contradiction. Fig. 6 depicts this situation. □

Proposition 4.3. *A language L is locally k-reversible if and only if L is recognized by $\tau_{k+1} \circ B$, where B is a reversible automaton.*

Proof. Proposition 4.1 proves that if L is recognized by $\tau_{k+1} \circ B$, where B is a reversible automaton, then L is locally k-reversible.

Let A be a locally k-reversible automaton that recognizes L. The transduction function τ_{k+1} is injective; therefore, any locally k-reversible language is recognized by $\tau_{k+1} \circ B$, where B is the reversible automaton defined in Proposition 4.2. □

Proposition 4.3 can be equivalently enunciated as follows:

Proposition 4.4. *A language L is locally k-reversible if and only if it is recognized by a semigroup in* $E^-_{com} * LI_{k+1}$.
The class of locally reversible languages (locally k-reversible for some value of k) corresponds with the variety $E^-_{com} * LI$. □

Note that $E^-_{com} * LI \subseteq LE^-_{com}$. In order to prove the reciprocal inclusion, to conclude that $E^-_{com} * LI = LE^-_{com}$, some previous results need to be proved.

Proposition 4.5. *Let L be a language recognized by a locally k-reversible automaton. The following conditions are fulfilled:*

(i) *for any words* v, w, z *and* u *such that* $|u| = k$*, if* $vu(wu)^+z \subset L$ *then* vuz *is in L;*
(ii) *for any words* v, w *and* u*, where* $|u| = k$*, there exist n and m such that*

$$u(vu)^n(wu)^m = u(wu)^m(vu)^n.$$

Proof. (i) Let $A = (Q, \Sigma, \delta, I, F)$ be a locally k-reversible automaton that accepts L. Let M be the transitions semigroup of A.

Let q_1 and q_2 be two states of the automaton such that they are in $\delta(Q, u)$, where $|u| = k$ (they are reached by the word u of length k), for any given word w, if $\delta(q_1, w)$ is equal to $\delta(q_2, w)$. then the states are the same one because the automaton is locally k-reversible.

Note that, given a word u of length k, then for each word w, $\delta(\delta(Q, w), u)$ is included into $\delta(Q, u)$. Therefore $\delta(Q, (wu)^i)$ is included into $\delta(Q, u)$ for any $i > 0$.

The semigroup M is finite; therefore there is a value n such that $\psi((wu)^n)$ is an idempotent in M. Thus, for any state q, it follows that $\delta(q, (wu)^n)$ is equal to $\delta(q, (wu)^{2n})$ and also to $\delta(\delta(q, (wu)^n), (wu)^n)$.

Note that $\delta(q, (wu)^n)$ is in $\delta(Q, u)$; thus, if the state q is in $\delta(Q, u)$, then $\delta(q, (wu)^n)$ is equal to q.

The automaton is trimmed; therefore, if q is in $\delta(Q, u)$, then there is a word v such that q is in $\delta(I, vu)$. If $vu(wu)^+z \subset L$, then $vu(wu)^nz$ is in L, concluding that vuz is in L because $\delta(q, (wu)^n)$ is equal to q.

(ii) Note that both $(vu)^n$ and $(wu)^m$ are subidentities on $rang(u)$, because for any word v there exists a value n such that $\psi(v^n)$ is an idempotent of the syntactic monoid of the language.

It is possible to formulate this same result in terms of the elements of $Syn(L)$; thus, for any given x, y in $Syn(L)$ and a word u such that $|u| = k$,

$$\psi(u)(x\psi(u))^\omega(y\psi(u))^\omega = \psi(u)(y\psi(u))^\omega(x\psi(u))^\omega,$$

where ψ denotes the syntactic morphism for the language. □

Proposition 4.6. *Let L be a language that fulfills the following conditions:*

(i) *if* $vu(wu)^+z \subset L$ *then* vuz *is in L;*
(ii) $u(vu)^n(wu)^m = u(wu)^m(vu)^n$*, for some values of n and m,*

where v, w, z *and* u *are words such that* $|u| = k$*, then L is locally k-reversible.*

Proof. To prove the proposition, we will show that $\tau_{k+1}(L)$ is in Rev, and therefore, by Proposition 4.1, L will be locally k-reversible. Let v, w, z and u be words with $|u| = k$.

(i) $vu(wu)^nz$ is in L, which implies that vuz is in L. Then, for any value of n, $\tau_{k+1}(vu(wu)^nz)$ in $\tau_{k+1}(L)$ implies that $\tau_{k+1}(vuz)$ is in $\tau_{k+1}(L)$. Therefore

$$\tau_{k+1}(vu)(\tau_{k+1}(uwu))^n\tau_{k+1}(uz)$$

in $\tau_{k+1}(L)$ implies that $\tau_{k+1}(vu)\tau_{k+1}(uz)$ is in $\tau_{k+1}(L)$.

(ii) Language L fulfills that $u(vu)^n(wu)^m = u(wu)^m(vu)^n$ for some values of n and m. Then, it is also fulfilled that $(\tau_{k+1}(uvu))^n(\tau_{k+1}(uwu))^m$ will be equal to $(\tau_{k+1}(uwu))^m(\tau_{k+1}(uvu))^n$.

Therefore, whenever L fulfills both conditions then $\tau_{k+1}(L)$ will be reversible ($\tau_{k+1}(L)$ is recognized by a monoid in E^-_{com}). Then, by Proposition 4.1, there will be a reversible automaton that accepts $\tau_{k+1}(L)$ and a locally k-reversible automaton that recognizes L. □

Propositions 4.5 and 4.6 allow us to characterize the class of locally k-reversible languages in the same way it is done by Pin for reversible languages [14].

Proposition 4.7. *Let L be a language. L is locally k-reversible if and only if it fulfills the following conditions:*

(i) *if* $vu(wu)^+z \subset L$ *then* vuz *is in L,*
(ii) $u(vu)^n(wu)^m = u(wu)^m(vu)^n$*, for some values of n and m,*

where v, w, z *and* u *are words and* $|u| = k$. □

The following proposition proves that $LE^{-}_{com} \subseteq E^{-}_{com} * LI$.

Proposition 4.8. *Let L be a language in LRev and ψ the syntactic morphism for the language. There exists a integer k such that, for each u in $\Sigma^{\geq k}$, and any words v, w, z, the following conditions are fulfilled:*

(i) *if $vu(wu)^{+}z \subset L$ then vuz is in L,*
(ii) $\psi(u)(x\psi(u))^{\omega}(y\psi(u))^{\omega} = \psi(u)(y\psi(u))^{\omega}(x\psi(u))^{\omega}$.

Proof. In any finite semigroup it is known that there exists an integer n such that $S^n = SE(S)S$ [16]. Let $u = u_1u_2u_3$ such that $\psi(u_2)$ is an idempotent of $Syn(L)$.

(i) Let $I \subseteq Syn(L)$ be an order ideal such that $\psi^{-1}(I) = L$.
$\psi(v)\psi(u)(\psi(w)\psi(u))^{\omega}\psi(z)$ is in I because $vu(wu)^{+}z$ is a proper subset of L. Then $\psi(v)\psi(u_1)e\psi(u_3)(\psi(w)\psi(u_1)e\psi(u_3))^{\omega}\psi(z)$ is in I as well as $\psi(v)\psi(u_1)\ (e\psi(u_3)\ \psi(w)\psi(u_1)e)^{\omega}\psi(u_3)\psi(z)$ because L is in $LRev$.
For each s in $Syn(L)$ and each idempotent e, $e \leq (ese)^{\omega}$, therefore

$$\psi(v)\psi(u_1)e\psi(u_3)\psi(z)$$

will be in I. Thus, we conclude that vuz is in L because $\psi(u_1)e\psi(u_3) = \psi(u)$.
(ii) In order to prove that $\psi(u)(v\psi(u))^{\omega}(w\psi(u))^{\omega} = \psi(u)(w\psi(u))^{\omega}(v\psi(u))^{\omega}$, we first consider the factorization of u, thus:

$$\psi(u)(v\psi(u))^{\omega}(w\psi(u))^{\omega} = \psi(u_1)e\psi(u_3)(v\psi(u_1)e\psi(u_3))^{\omega}(w\psi(u_1)e\psi(u_3))^{\omega},$$

from which follows

$$\begin{aligned}\psi(u_1)(e\psi(u_3)v\psi(u_1)e)^{\omega}e\psi(u_3)(w\psi(u_1)e\psi(u_3))^{\omega} &= \psi(u_1)(e\psi(u_3)v\psi(u_1)e)^{\omega}(e\psi(u_3)w\psi(u_1)e)^{\omega}\psi(u_3)\\ &= \psi(u_1)(e\psi(u_3)w\psi(u_1)e)^{\omega}(e\psi(u_3)v\psi(u_1)e)^{\omega}\psi(u_3)\end{aligned}$$

and finally

$$\psi(u)(w\psi(u_1)e\psi(u_3))^{\omega}(v\psi(u_1)e\psi(u_3))^{\omega} = \psi(u)(w\psi(u))^{\omega}(v\psi(u))^{\omega}. \quad \square$$

From $E^{-}_{Com} * LI = LE^{-}_{Com}$ there follows the decidability of locally reversible languages. In order to do so, it suffices to first, compute the syntactic order semigroup; second, compute all the local submonoids; and finally, check if all of them belong to E^{-}_{com}. In other words, it suffices to check if the idempotents commute; and, if for any idempotent e we have $1 \leq e$.

By Proposition 4.1 we can decide if a certain language is locally k-reversible, it suffices to compute $\tau^{-1}_{k+1}(L)$ and check if it is reversible. Besides, for any given locally reversible language, it is possible to compute the smallest k such that the language is locally k-reversible. Note that it is enough to iterate over a process that conjecture a value of k (initially zero) and check if the language L is locally k-reversible. Since we already know that L is locally reversible we will find the minimal k such that L is locally k-reversible.

5. Construction of a quasi-*k*-reversible automaton

First, we recall the definition of quasi-reversible automata [12].

Definition 5.1 (*Lombardy '02*). An automaton $A = (Q, \Sigma, \delta, I, F)$ is said to be quasi-reversible if the following conditions are fulfilled:

1. Given two transitions (p, a, q) and (p, a, r), none of them belong to an SCC.
2. Given two transitions (p, a, r) and (q, a, r), none of them belong to an SCC.

Quasi-reversible automata are the most suitable representation, in terms of space efficiency, for the class of reversible languages. With respect to reversible automata, quasi-reversible automata have similar properties and can be easily processed to obtain reversible ones. We now extend the quasi-reversibility conditions in order to deal with locally k-reversible languages.

Definition 5.2. An automaton $A = (Q, \Sigma, \delta, I, F)$ is said to be quasi-k-reversible if the following conditions are fulfilled:

1. Given two transitions (p, a, q) and (p, a, r), none of them belong to an SCC.
2. Given two transitions (p, a, r) and (q, a, r), if any of them is into an SCC, then there is no word u of length k that reaches both p and q.

We will refer to them as the first and second quasi-k-reversibility conditions. The next proposition extends a previous result concerning reversible languages [6].

Proposition 5.3. *Let L be a locally k-reversible language and A a DFA that accepts L. Whenever, for some u in Σ^k and some symbol a, there are transitions (p, a, r) and (q, a, r) together with paths (p', u, p) and (q', u, q), where (q', u, q) and (q, a, r) are into an SCC, then $q < p$.*

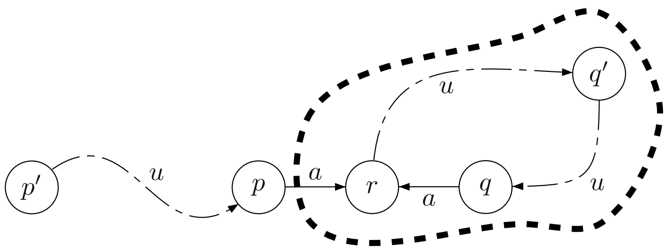


Fig. 7. DFA for a locally k-reversible language with two states p and q that have transitions with the same symbol to the same state. If there is a word u of length k reaching both states, and q is into an SCC, then $q < p$.

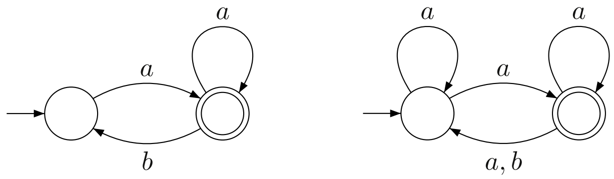


Fig. 8. The figure on the left shows the minimal DFA for the language $a(a + ba)^*$. The universal automaton A for the same language is shown on the right.

Proof. Let z be a word z such that $\delta(q, z)$ is in F. Then there exists a word y such that $\delta(p, (ayu)^i z)$ is in F for any integer i.

If $\delta(q_0, x) = p'$ then $xu(ayu)^i z$ is in L for any integer $i \geq 1$. The language L is locally k-reversible; therefore xuz is also in L and $\delta(p, z)$ is in F. Fig. 7 illustrates the proof. □

In [6] we give an algorithm to obtain a quasi-reversible automaton from the minimal DFA of a given reversible language. The algorithm obtains the saturated RFSA of the minimal DFA and erases all the transitions that violate any quasi-reversibility condition.

When locally k-reversible languages are considered, this algorithm (taking into account the conditions of quasi-k-reversibility) cannot be applied. When reversible languages are considered, the SCCs of the universal automaton are reversible, and therefore the SCCs of the saturated RFSA of the minimal DFA are also reversible. This is not true when k is greater than zero. As an example, let us consider the language $L = a(a + ba)^*$ which is locally 1-reversible. The minimal DFA for the language and the universal automaton are in Fig. 8. Note that in both cases the automata have only one SCC; the minimal DFA is locally k-reversible; the universal automaton is not locally reversible.

The next result proves that SCCs of the minimal DFA are locally k-reversible. This result extends a previous one concerning SCCs of the universal automaton of a reversible automaton [12].

Proposition 5.4. *Let L be a locally k-reversible language and let the automaton $A = (Q, \Sigma, \delta, q_0, F)$ be the minimal DFA for the language. The SCCs of the automaton are locally k-reversible.*

Proof. Let consider the situation depicted in Fig. 9 that does not fulfill the definition of local k-reversibility.

Let x, z be words such that $\delta(q_0, x) = p_1$ and $\delta(q_2, z)$ is in F. The words $xu(aw_2u)^+z$ are in L, and therefore also the word xuz. This implies that $\delta(q_1, z)$ is in F.

Analogously, let x be a word such that $\delta(q_0, x) = p_2$ and z a word such that $\delta(q_1, z)$ is in F; in the same way it follows that $\delta(q_2, z)$ is in F. Therefore, the states q_1 and q_2 are equivalent and the configuration shown in Fig. 9 is impossible because the automaton A is minimal. □

We now describe the process to obtain a quasi-k-reversible automaton for any locally k-reversible language L. We consider the minimal DFA $A = (Q, \Sigma, \delta, q_0, F)$ for the language, and initially, we partially saturate the DFA to obtain an RFSA A^*. Initially, we set the RFSA to $A^* = (Q, \Sigma, \delta^*, I, F)$, where δ^* is equal to δ and $I = \{q_0\}$.

Let δ_2 be the set of transitions of A that violate the second quasi-k-reversibility condition. For each transition (p, a, r) in δ_2 there exists another transition (q, a, r) into an SCC (note that this implies that $q < p$).

For each transition (p, a, r) in δ_2 and each transition (s, b, p) in δ, add a new transition (s, b, q) to δ^*. The new transition is also added to δ_2 whenever it violates the second quasi-k-reversibility condition. Besides, the state q is added to I when the state p is in I. This process ends when all the transitions in δ_2 have been considered. The following proposition proves that the automaton A^* recognizes L.

Proposition 5.5. *The partially saturated automaton A^* described above recognizes the language L.*

Proof. Initially, $A^* = (Q, \Sigma, \delta, \{q_0\}, F)$, and trivially $L(A^*) = L$.

For each transition (p, a, r) in δ_2 and each transition (s, b, p) in δ, the algorithm adds a new transition (s, b, q) to δ^*. Note that the new transitions added do not modify the language accepted by A^* because $q < p$. □

We now prove that, taking into account the RFSA A^*, it is possible to delete the transitions that violate either the first or the second quasi-k-reversibility condition in order to obtain an equivalent quasi-k-reversible automaton for the language.

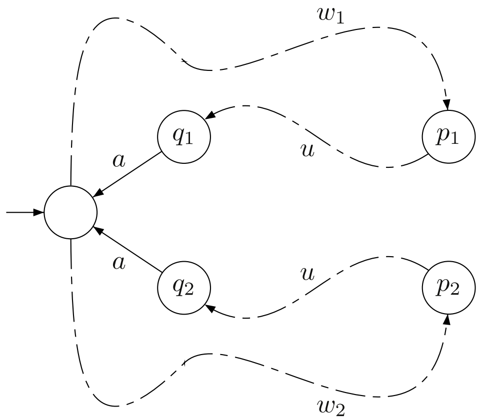


Fig. 9. For any locally k-reversible language given, the scheme shows a forbidden configuration of the minimal DFA of the language.

Proposition 5.6. *Let L be a locally k-reversible language and let A^* be the partially saturated RFSA obtained as explained above. Let A' be the automaton obtained when all the transitions of A^* that violate the second quasi-k-reversibility condition are erased. The automaton A' accepts the language L.*

Proof. Let δ_2 denote the set of transitions of A^* that violate the second quasi-k-reversibility condition. We will prove by induction that, for each word x, the set of reachable states in A^* from I is the same when the transitions in $\delta^* - \delta_2$ are exclusively considered.

If $|x| = 0$ the set is I. Let us suppose that the proposition is fulfilled for $|x| \leq k$. Let $x = x'a$ with $|x'| = k$. If the state r is reached by x, in such a way that there is a state p in $\delta(I, x')$ and a transition (p, a, r) in δ_2, this implies that there exists a transition (q, a, r) into an SCC such that $q < p$. Therefore, q is in $\delta(I, x')$ and the state r can be reached without using the transition (p, a, r). □

Proposition 5.7. *Let L be a locally k-reversible language and let A' be the automaton obtained by following the guidelines of Proposition 5.6. Given any word $x \in \Sigma^*$ that reaches the state q from some initial state, there is another state p, such that $q < p$, that is reachable by x from some initial state and using transitions that do not violate the first quasi-k-reversibility condition.*

Proof. Let δ_1 denote the set of transitions of A' that violate the first quasi-k-reversibility condition. We will prove the proposition by induction on the length of x.

The base of induction, $|x| = 0$, is trivially proved. Let us suppose that the proposition is fulfilled when $|x| \leq k$.

Now let $x = x'a$, where $|x'| = k$, and such that it reaches the state q from some initial state. There will also exist a state r such that it is reachable by x' together with a transition (r, a, q).

Two situations arise depending on whether the transition (r, a, q) belongs to δ_1 or not. If the transition (r, a, q) is in δ_1, then there will exist another transition (r, a, p) inside an SCC. On the one hand, if x' reaches the state r without using transitions in δ_1, then it is clear that $x'a$ reaches p such that $p > q$. On the other hand, if x' reaches r but using transitions in δ_1, then x' will also reach a state r' such that $r' > r$. The relation $>$ is a right congruence; thus, there exist a state p' and a transition (r', a, p') such that p' is reached by $x'a$ and $p' > p$ because of the existence of the transition (r, a, p). The transitivity of the relation $>$ implies that $p' > q$.

If the transition (r, a, q) is not in δ_1 the proposition is proved. □

Taking into account the automaton A', let A'' be the automaton obtained when all the transitions that violate the first quasi-k-reversibility condition are deleted. The following corollary proves that A'' accepts the same language and that the automaton is quasi-k-reversible.

Corollary 5.8. *Let L be a locally k-reversible language and let A' be the automaton obtained described in Proposition 5.6. Let the automaton A'' be the resulting automaton from erasing all the transitions in A'' that violate the first quasi-k-reversibility condition.*

The automaton A'' recognizes L and it is quasi-k-reversible.

Proof. Proposition 5.7 proves that, for any word x reaching a final state f_1, whenever transitions that violate the first quasi-k-reversibility condition are used, the word also reaches another final state f_2 such that $f_1 < f_2$. Therefore, $L(A'') = L(A')$.

Trivially, $L(A'') = L$ and the automaton A'' is quasi-k-reversible. □

Propositions 5.6 and 5.7 prove that it is feasible to erase the transitions that violate some quasi-k-reversibility condition from the partially saturated RFSA to obtain an equivalent quasi-k-reversible automaton. Algorithm 1 summarizes the process. An example of a run is shown in Example 5.9.

Algorithm 1 Algorithm to obtain a quasi-k-reversible automaton for any locally k-reversible language.

Input: Minimal DFA $A = (Q, \Sigma, \delta_A, q_0, F)$ of a locally k-reversible language L
Output: Quasi-k-reversible automaton $B = (Q, \Sigma, \delta_B, I, F)$ that accepts L

Method
// Initialization
$\delta_B = \delta_A$; $I = \{q_0\}$

Let δ_2 be the set of transitions that violate the second quasi-k-reversibility condition;
$\delta_B = \delta_B - \delta_2$;
for each production (p, a, r) in δ_2:
  take into account that there is a transition (q, a, r) in δ_A;
  for each transition (s, b, p) in δ_A:
    if the new transition (s, b, q) violates no quasi-k-reversibility condition:
      add the transition (s, b, q) to δ_B;
    endif
    if the transition (s, b, q) violates the second quasi-k-reversibility condition and it is not already in δ_2:
      add the transition (s, b, q) to δ_2;
    endif
    if the state p is in I:
      add the state q to I;
    endif
  endfor each
endfor each
EndMethod

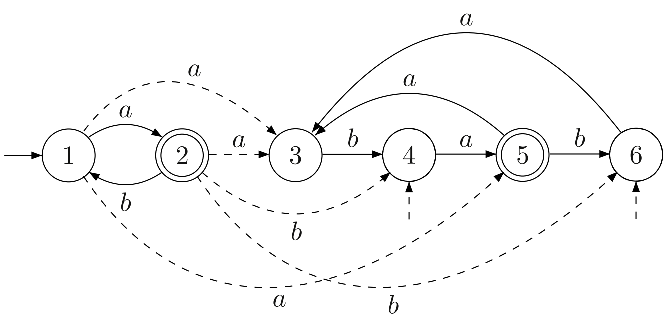


Fig. 10. Partially saturated RFSA obtained after applying the transformation described in Proposition 5.5. Transitions in δ_2 and new initial states are marked with a dash.

Example 5.9. Let consider the language accepted by the locally 1-reversible automaton shown in Fig. 2. Let $A = (Q, \Sigma, \delta_A, q_0, F)$ be the minimum DFA for the language. It is shown in Fig. 3. Note that the minimum DFA is neither locally 1-reversible nor quasi-1-reversible. Let $B = (Q, \Sigma, \delta_B, I, F)$ be the quasi-k-reversible automaton, where initially δ_B is set to δ_A and $I = \{1\}$.

Note that only the transition $(2, a, 3)$ violates the second quasi-k-reversibility condition (initially $\delta_2 = \{(2, a, 3)\}$). The relation $5 < 2$ is detected but the transition $(1, a, 5)$ is not added to δ_B because the transition violates the both the quasi-k-reversibility conditions (it is added to δ_2).

The consideration of the transition $(1, a, 5)$ leads to detecting the relation $4 < 1$ and therefore the algorithm includes the state 4 in the set of initial states. The transition $(2, b, 4)$ is not included in δ_B but in δ_2 because it also violates the second quasi-k-reversibility condition. The treatment of this transition leads to establishing the relation $3 < 2$ and to the inclusion of the transition $(1, a, 3)$ in δ_2 because it violates both quasi-k-reversibility conditions.

When the transition $(1, a, 3)$ is considered, it leads to $6 < 1$ and, therefore, to the inclusion of the state 6 in the set of initial states.

The algorithm also considers the inclusion of the transition $(2, b, 6)$. It is included in δ_2 to be treated. It leads to the relation $5 < 2$ already detected; thus no transition is added, and the process ends. Fig. 10 shows the resulting automaton.

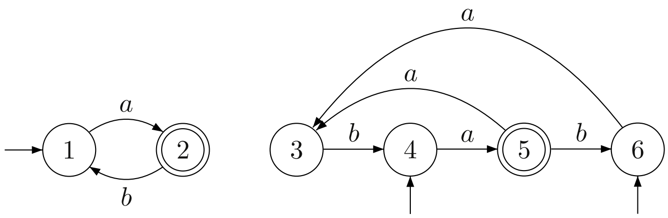


Fig. 11. Quasi-1-reversible automaton obtained.

Note that none of the transitions that violate the quasi-k-reversibility conditions are included but treated. The final automaton obtained is shown in Fig. 11. □

Note that the main loop of Algorithm 1 considers all the transitions that violate the second quasi-k-reversibility condition (transitions in δ_2). Note that the number of these transitions is bounded by the number of transitions of the saturated RFSA for the minimal DFA of the language ($\mathcal{O}(n^2|\Sigma|)$, where n denotes the size of the minimal DFA).

For each of the transitions in δ_2, the algorithm creates a transition to the *smaller* state q for each transition that reaches the *bigger* state p. Therefore, the number of transitions to add is bounded by the number of transitions in the minimal DFA ($\mathcal{O}(n|\Sigma|)$.

These new transitions are also processed to test their ownership to δ_2. This implies testing if there exists a common word of length k reaching both the states p and q. This can be computed by intersecting the reverse automata with initial states p and q (with time complexity of $\mathcal{O}(n^2)$). Using this intersection automaton, it is enough to check whether there exists or not a path of length greater than or equal to k. Thus, the time complexity of testing the ownership of the transition to δ_2 is $\mathcal{O}(n^3|\Sigma|k)$.

In the worst case, all but one of the transitions in δ_2 are processed inside the main loop; therefore, the time complexity of the algorithm is bounded by $\mathcal{O}(n^5|\Sigma|^3k)$.

An anonymous referee suggested an alternative algorithm to compute the quasi-k-reversible automaton.

Given an automaton for any language L, the algorithm consists in computing a quasi-reversible automaton for the language $\tau_{k+1}(L)$. The quasi-reversible automaton is then composed with the inverse transducer τ_{k+1}^{-1} to obtain a quasi-k-reversible automaton.

The referee questions whether the resulting automaton is bigger than the automaton obtained by applying Algorithm 1 or not. With respect to this, we note that both the transductions to be performed cannot reduce the number of states of the automaton; therefore, the size of the automaton has the size of the minimal DFA for the language as a lower bound (as well as the size of the automaton obtained by Algorithm 1). Besides, we also note that the size of the transducer is exponential with respect to k. Therefore, there might be some cases with bigger results.

6. Conclusions

In this work we define the class of locally k-reversible languages as the union closure of k-reversible languages [1]. This work is motivated by previous work by Pin [13] that study the class of reversible languages as the union closure of Angluin's 0-reversible languages.

The class of locally k-reversible languages is algebraically characterized and it is proved that $LE_{com}^{-} = E_{com}^{-} * LI$. In order to properly represent the defined class, the definition of quasi-reversible automata is extended. An algorithm is also proposed to obtain the quasi-k-reversible automaton from the minimal DFA of a locally reversible language. The size of the quasi-k-reversible automaton is bounded by the number of states of the minimal DFA. The quasi-k-reversible automaton can be easily modified to obtain a collection of locally k-reversible automata. This result extends previous papers on reversible automata and languages [12,6].

Acknowledgement

The authors thank the anonymous referee for his thoughtful comments that really helped to improve the initial manuscript.

Appendix. An alternative definition of locally *k*-testable automata

An alternative, more symmetric, definition for locally k-testable automata was suggested by an anonymous referee. It is enunciated below:

Definition A.1. An automaton is locally (k, m)-reversible if the following conditions hold:

- if there exist two transitions (p, a, r) and (q, a, r) in the automaton, then there is no word u of length k that reaches both p and q;
- if there exist two transitions (r, a, p) and (r, a, q) in the automaton, then there does not exist a path from both states p and q labelled by the same word u of length m.

Note that, in this framework, a reversible automaton is a locally $(0, 0)$-reversible automaton.

In order to prove the equivalence of this alternative definition and Definition 3.1 we will use both of them simultaneously. In order to avoid confusion, we comment to the reader that Definition 3.1 considers just one argument but Definition A.2 takes into account two arguments.

Definition A.2. Given an alphabet Σ, let us define the function $t_k : \Sigma^* \to (\Sigma^k)^*$ as follows:

$$t_k(a_1a_2\dots a_m) = \begin{cases} \lambda & \text{if } m < k \\ [a_1\dots a_k][a_2\dots a_{k+1}]\dots[a_{m-k+1}\dots a_m] & \text{otherwise} \end{cases}$$

The function t_k can be extended to languages in the usual way; that is:

$$L_k = \{t_k(x) \ : \ x \in L\}.$$

In an intuitive way, for any given word u (resp. language L) over Σ, the function $t_k(u)$ returns the word of k-segments of u (resp. the language of k-segment words).

Note that we can relate the above definition together with Propositions 4.1 and 4.2 and state that, for any language L, it is locally k-reversible if and only if L_{k+1} is reversible. We now prove that Definitions 3.1 and A.2 are equivalent.

Proposition A.3. *Let $A = (Q, \Sigma, \delta, I, F)$ be an automaton. It is locally (k, m)-reversible if and only if it is locally $(k + m)$-reversible.*

Proof. On the one hand, note that any locally k-reversible automaton is locally $(k, 0)$-reversible.

On the other hand, in order to prove that any locally (k, m)-reversible automaton A is locally k-reversible, we will prove that the language $L(A)_{m+k+1}$ is reversible.

Note that, if L is locally (k, m)-reversible, then the language $\tau_{k+1}(L)$ is locally $(0, m)$-reversible. Let us consider now the reverse of this language, which we will denote by $(\tau_{k+1}(L))^r$. Also, note that this language is locally $(m, 0)$-reversible.

Note that the language $\tau_{m+1}((\tau_{k+1}(L))^r)$ is locally $(0, 0)$-reversible, as well as the language $(\tau_{m+1}((\tau_{k+1}(L))^r))^r$.

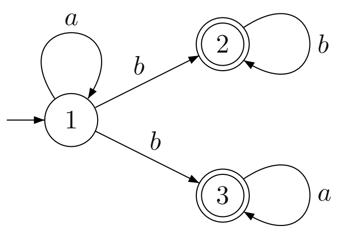
From $(\tau_{k+1}(L))^r = \tau_{k+1}(L^r)$ it follows that

$$\begin{aligned}(\tau_{m+1}((\tau_{k+1}(L))^r))^r &= \tau_{m+1}((\tau_{k+1}(L))^r)^r = \\ &= \tau_{m+1}(\tau_{k+1}(L)) = \tau_{m+k+1}(L)).\end{aligned}$$

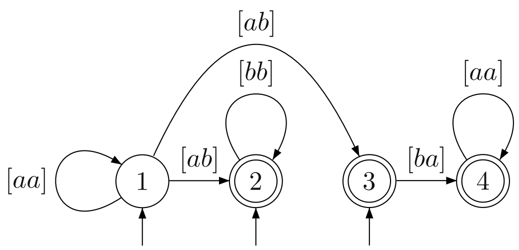
Please note that the language $\tau_{m+k+1}(L))$ (i.e. $L(A)_{m+k+1}$) is reversible, and therefore the language L is locally $(k + m)$-reversible. □

We now give an example that illustrates Proposition A.3.

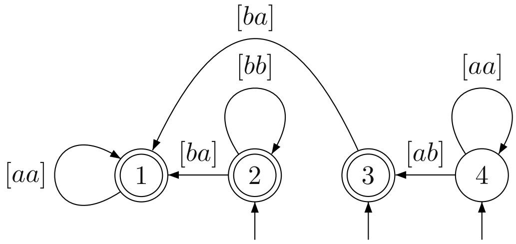
Example A.4. Let L be the language accepted by the following locally $(1, 1)$-reversible automaton:



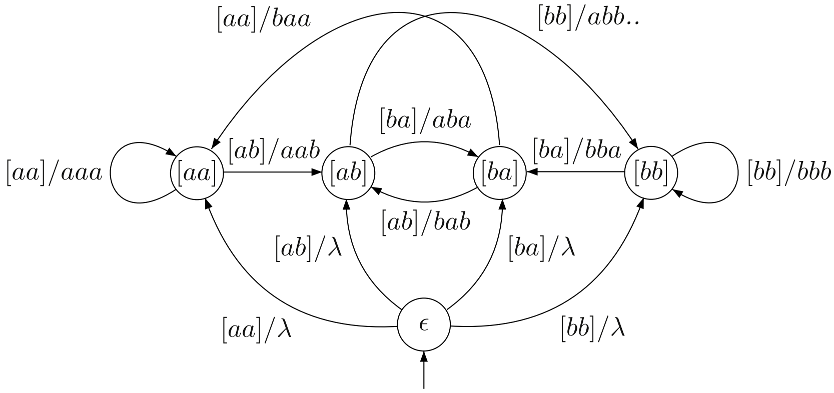
Taking into account the transducer τ_2 shown in Fig. 4, we take into account the construction in Proposition 4.2 in order to obtain the locally $(0, 1)$-reversible automaton for $\tau_2(L)$:



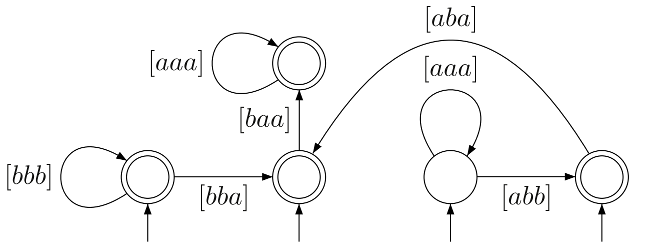
Note that the automaton for the language $(\tau_2(L))^r$ is locally $(1, 0)$-reversible. This automaton is shown below:



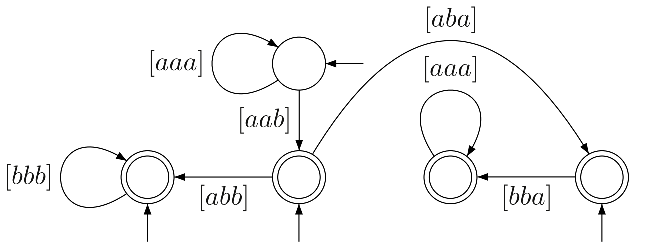
In order to operate with this automaton, it is necessary to take into account the transducer whose input and output alphabets are respectively Σ^2 and Σ^3. This transducer (τ_2') is now shown.



The construction in Proposition 4.2 is used again to obtain $\tau_2'((\tau_2(L))^r)$. Note that the resulting automaton is reversible:



The reverse of this automaton accepts the language L_3. This automaton is now shown:



The automaton is reversible; therefore, by Proposition A.3, the language L is locally 2-reversible. $\square$

References

[1] D. Angluin, Inference of reversible languages, Journal of the ACM 29 (3) (1982) 741–765.
[2] F. Denis, A. Lemay, A. Terlutte, Residual finite state automata, Fundamenta Informaticae 51 (4) (2002) 339–368.
[3] F. Denis, A. Lemay, A. Terlutte, Learning regular languages using non-deterministic finite automata, Lecture Notes in Computer Science 1891 (2004) 39–50.
[4] F. Denis, A. Lemay, A. Terlutte, Learning regular languages using rfsa, Theoretical Computer Science 313 (2) (2004) 267–294.
[5] S. Eilenberg, Automata, Languages and Machines, vol. B, Academic Press, New York, 1976.
[6] P. García, M. Vázquez de Parga, D. López, On the efficient construction of quasi-reversible automata for reversible languages, Information Processing Letters 107 (1) (2008) 13–17.
[7] P. García, J. Ruiz, Learning in varieties of the form $V * LI$ from positive data, Theoretical Computer Science 362 (1–3) (2006) 100–114.
[8] A. Gingsburg, Algebraic theory of automata, in: ACM Monograph Series, Academic Press, 1968.

[9] M. Golovkins, J.E. Pin, Varieties generated by certain models of reversible automata, unpublished manuscript.
[10] T. Hall, Biprefix codes, inverse semigroups and syntactic monoids of injective automata, Theoretical Computer Science 32 (1984) 201–213.
[11] P.-C. Héam, A lower bound for reversible automata, Theoretical Informatics and Applications 34 (2000) 331–341.
[12] S. Lombardy, On the construction of reversible automata for reversible languages, in: Proceedings of the 29th ICALP, Lecture Notes in Computer Science 2380 (2002) 170–182.
[13] J.-E. Pin, On the languages recognized by finite reversible automata, in: Proceedings of the 14th ICALP, Lecture Notes in Computer Science 267 (1987) 237–249.
[14] J.-E. Pin, On reversible automata, in: Proceedings of the first LATIN conference, Lecture Notes in Computer Science 583 (1992) 401–416.
[15] J.-E. Pin, A variety theorem without complementation, Russian Mathematics (Izvestija vuzov.Matematika) 39 (1995) 80–90.
[16] J.E. Pin, Varieties of Formal Languages, North Oxford Academic Publishers Ltd., 1986.
[17] P.V. Silva, On inverse monoid languages, Theoretical Informatics and Applications 30 (1996) 349–378.