Theoretical Computer Science 410 (2009) 16–34

Contents lists available at ScienceDirect

Theoretical Computer Science

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

Finite n-tape automata over possibly infinite alphabets: Extending a theorem of Eilenberg et al.

Christian Choffrut *, Serge Grigorieff

LIAFA, Université Paris 7, 2, pl. Jussieu 75251 Paris Cedex 05, France
CNRS, Université Paris 7, 2, pl. Jussieu 75251 Paris Cedex 05, France

ARTICLE INFO

Article history:
Received 20 March 2006
Received in revised form 27 June 2008
Accepted 23 July 2008
Communicated by Z. Esik

Keywords:
Finite automata
First order logic
Synchronous relations
Infinite alphabets

ABSTRACT

Eilenberg, Elgot and Shepherdson showed in 1969, [S. Eilenberg, C.C. Elgot, J.C. Shepherdson, Sets recognized by n-tape automata, Journal of Algebra 13 (1969) 447–464], that a relation on finite words over a finite, non-unary alphabet with p letters is definable in first order logic with $p+2$ predicates for the relations *equal length*, *prefix* and *last letter is a* (for each letter $a \in \Sigma$) if and only if it can be recognized by a finite multitape synchronous automaton, i.e., one whose read heads move simultaneously. They left open the characterization in the case of infinite alphabets, and proposed some conjectures concerning them. We solve all problems and sharpen the main theorem of [S. Eilenberg, C.C. Elgot, J.C. Shepherdson, Sets recognized by n-tape automata, Journal of Algebra 13 (1969) 447–464].

1. Introduction

The purpose of this work is

(A) to propose a notion of multitape synchronous automaton for an infinite alphabet,
(B) to answer a few related questions raised by Eilenberg et al
(C) to sharpen their main theorem with restrictions of the logical language.

The answers we give to questions (B) somehow suggest that our proposal for (A) is robust.

For the reader familiar with the theory of rational relations over free finitely generated monoids, as exposed in several textbooks such as [3,8,12], we recall that the importance of the family of synchronous relations is due to the fact that they are a good trade-off between
- the general family of rational relations which have high expressive power but for which very few properties are decidable and most closure properties fail,
- and the subfamily of recognizable relations with rich closure and decidability properties but weak expressive power.

It is not surprising that this is the family which is most often considered in several applications, e.g., in database theory [2,5], model checking [13,7] and in automatic group theory [10].

1.1. The problems left open by Eilenberg & al.

We consider a few questions raised by Eilenberg, Elgot, Shepherdson in [9]. Their main result is a characterization, in terms of n-tape finite automata whose reading heads move synchronously and which we shall call EES-automata, of the

* Corresponding author. Tel.: +33 1 44 27 79 84; fax: +33 1 44 27 68 49.
E-mail addresses: Christian.Choffrut@liafa.jussieu.fr (C. Choffrut), Serge.Grigorieff@liafa.jussieu.fr (S. Grigorieff).
URLs: http://www.liafa.jussieu.fr/~cc (C. Choffrut), http://www.liafa.jussieu.fr/~seg (S. Grigorieff).

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2008.07.018

n-ary relations on words (i.e., subsets of the direct product $\Sigma^* \times \cdots \times \Sigma^*$) which are definable in the first order theory of the free monoid Σ^* with

- the binary predicate `Pref` which means "*x is a prefix of y*",
- the binary predicate `EqLen` which means "*x and y are of equal length*"
- and the unary predicates $\mathtt{Last}_a$ which mean "*x ends with the letter a*" (one predicate for each letter $a \in \Sigma$).

This result holds only when the number of letters is finite and greater than one. Simple counter-examples are given which show that the automata model is strictly more powerful than the logic when the number of letters is equal to one (cf. [9, Theorem 9.1]) or when it is infinite (cf. [9, Section 10, Example 1]). When the alphabet is finite, EES-automata are exactly what is now called synchronous automata. In case the alphabet is infinite, the transitions of EES-automata involve *arbitrary* (possibly non computable) sets of n-tuples of symbols, a rather surprising feature for automata. This is why Eilenberg & al. suggest ([9, Section 10, Problem 2]) to restrict the notion of relation recognizable by EES-automata to relations on words which are invariant under all the permutations of Σ which act as the identity on some finite subalphabet Σ_0. Let us call such relations Σ_0-*finitary*. Eilenberg & al. state three open problems for the case of an infinite alphabet.

- Problem 1. Are the binary relation `EqLenEqLast` which means "*u and v have the same length and end with the same letter*" and its restriction $\{(xz, yz) \mid x, y, z \in \Sigma\}$ to words of length 2 definable with `Pref`, `EqLen` and the $\mathtt{Last}_a$'s?

There exist very simple automata recognizing the relation `EqLenEqLast`, so more generally they ask

- Problem 2. If a relation is recognized by an EES-automaton and is Σ_0-finitary for some subset $\Sigma_0 \subseteq \Sigma$, is it definable with `Pref`, `EqLen` and the $\mathtt{Last}_a$'s?

The special case where Σ_0 is empty leads to the last question which does not involve automata

- Problem 3. If a relation is definable with `Pref`, `EqLen` and the $\mathtt{Last}_a$'s and is invariant under all permutations of Σ, is it definable without the $\mathtt{Last}_a$ predicates?

Let us add to these problems the following refinement of Problem 3.

- Problem 3bis. If a relation is definable with `Pref`, `EqLen` and the $\mathtt{Last}_a$'s and is Σ_0-finitary for some subset $\Sigma_0 \subseteq \Sigma$, is it definable with `Pref`, `EqLen` and the $\mathtt{Last}_a$'s, for a varying in Σ_0 ?

1.2. *Our contribution*

We clarify the relation between automata and the above logic and solve all three problems. Also, we consider the logic obtained by restricting the predicates $\mathtt{Last}_a$'s to $a \in \Sigma_0$, a finite subalphabet of Σ. Except for a few of them, all results are valid for both finite and infinite alphabets Σ.

This can be summarized as follows (cf. also the table in Fig. 1).

1. We impose on the transitions of EES-automata the Σ_0-finitary condition. Indeed, we show that, for words over an infinite alphabet, a relation recognized by some EES-automaton satisfies the Σ_0-finitary condition if and only if so do all of its transitions, cf. Theorem 3.4. We thus obtain a reasonable notion of automata for infinite alphabets which we call Σ_0-*synchronous* and which also makes sense for finite alphabets. Concerning the relationship between automata and logic, we prove that the relations recognized by these automata are exactly those definable with the predicates `Pref`, `EqLen` and $\mathtt{Last}_a$ for $a \in \Sigma_0$ along with the predicate `EqLenEqLast` (meaning "*same length and same last letter*"), cf. Theorem 4.1. This result is valid for both finite and infinite alphabets.

2. We introduce the subclass of *oblivious* Σ_0-synchronous automata and characterize the family of relations recognized by such automata in terms of EES-automata and saturation under a suitable equivalence involving Σ_0, cf. Theorem 3.12. When $\Sigma_0 \neq \emptyset$, we prove that the relations recognized by these automata are exactly those definable with the predicates `Pref`, `EqLen` and $\mathtt{Last}_a$ for $a \in \Sigma_0$, cf. Theorem 4.3. This leads to a positive answer to Problem 3bis, cf. Corollary 4.4. As a by-product, this also leads to a negative answer to Problem 1, cf. Corollary 4.4, and therefore to Problem 2. The particular case $\Sigma_0 = \emptyset$ must be handled differently. We prove that the relations recognized by our notion of constant-free oblivious synchronous automata are exactly those definable with the predicates `Pref`, `EqLen` and the unary predicates $\mathtt{mod}_{k,\ell}$ meaning "*the length of u is congruent to k modulo* ℓ", cf. Theorem 4.5. This solves Problem 3, see Section 4.5. Also, we prove that the relations recognized by *non-counting* constant-free oblivious synchronous automata are exactly those definable with the sole predicates `Pref` and `EqLen`, cf. Theorem 4.8. Finally, we show quantifier elimination in the logics $(\mathtt{Pref}, \mathtt{EqLen}, (\mathtt{mod}_{k,\ell})_{k<\ell})$ and $(\mathtt{Pref}, \mathtt{EqLen})$ for simple extensions of these languages, cf. Propositions 4.6 and 4.9.

The reader will find some common flavor between the notion of oblivious synchronicity and that of "regular prefix relation" due to Angluin and Hoover, 1984 [1], rediscovered by Laüchli and Savioz, 1987 [11] (cf. Choffrut, 2006 [6]). Nevertheless, the two notions are not comparable, cf. Section 4.7.

Automata for recognition	Logics on Σ^* for definability	Relations
EES	see Alexis Bès [4]	right invariant congruence of finite index (†)
Σ_0-synchronous	$\mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a\in\Sigma_0}$, $\mathtt{EqLenEqLast}$	(†) + Σ_0-finitary (*)
		(†) + $\equiv^{\mathrm{sync}}_{n,\Sigma_0}$-saturated
oblivious Σ_0-synchronous (if $\Sigma_0 \neq \emptyset$)	$\mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a\in\Sigma_0}$	(†) + $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturated
constant-free oblivious synchronous	$\mathtt{Pref}, \mathtt{EqLen}, (\mathtt{mod}_{k,\ell})_{k<\ell}$	(†) + $\equiv^{\mathrm{obl}}_{n,\emptyset}$-saturated
non-counting oblivious constant-free synchronous	$\mathtt{Pref}, \mathtt{EqLen}$	(†) + non-counting + $\equiv^{\mathrm{obl}}_{n,\emptyset}$-saturated

Fig. 1. Automata and logic for relations on words. (*) is valid only if Σ is infinite.

3. We prove that adding generalized quantifiers $\exists^\infty$ (meaning *there are infinitely many solutions*) and $\exists^{k \bmod \ell}$ (meaning *the number of solutions is finite and congruent to k modulo* ℓ) does not extend the expressive power of $(\mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a\in\Sigma_0})$ and $(\mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a\in\Sigma_0})$, cf. Theorem 4.10.

4. From these results, we obtain (cf. Theorem 4.11) the decidability of the first order theory (with quantifiers $\exists^\infty$ and $\exists^{k \bmod \ell}$ allowed) of the structure

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a\in\Sigma}\rangle.$$

The paper is organized as follows. In Section 2 we discuss the problem of extending the notion of finite automata to possibly infinite alphabets, which we call EES automata after the three authors Eilenberg, Elgot and Shepherdson, and study the particular case where the labels, which are subsets of the alphabet, are invariant under all permutations fixing all the elements of some finite subset. This allows us to define in Section 3 the family of synchronous finite automata over infinite alphabets, and the subfamily of oblivious synchronous automata and the corresponding relations. We investigate their general closure properties, and give characterizations in terms of special equivalences which are used in the last section. In Section 4 we are concerned with the different logical structures using the natural elementary predicates, such as those in the table in Fig. 1 and prove the equivalence between families of automata and families of logics as suggested by the first two columns of the table.

2. Finite automata over an infinite alphabet

The purpose of this section is to look at various notions of finite n-tape automata that can be used to recognize n-tuples of words over infinite alphabets, and to establish some properties of the class of relations over words that they define. The ultimate goal is to establish a correspondence between these families of relations and those definable in the diverse logics introduced in Section 4.

2.1. Words

Let Σ be a finite or infinite alphabet and let Σ^* be the free monoid it generates, i.e., the set of all finite sequences of elements in Σ, also called *words*. The *length* of $u \in \Sigma^*$ is denoted by $|u|$. For $1 \le k \le |u|$, $u[k]$ denotes the k-th letter of u and $u \upharpoonright k$ the prefix of u of length k. We denote by ε the empty word, i.e., the word of length 0. The *concatenation* product of two words u and v is denoted by uv, so that u (resp. v) is a *prefix* (resp. *suffix*) of the word uv.

Given an integer $n > 0$, the direct product $\overbrace{\Sigma^* \times \cdots \times \Sigma^*}^{n \text{ times}}$ has the structure of a monoid with componentwise concatenation. Considering a new symbol # not in Σ, we pad all short components of any n-tuple $(w_1, w_2, \ldots, w_n) \in (\Sigma^*)^n$ with as few occurrences of # as necessary to make the length of all components the same:

$$\begin{array}{l}(w_1, w_2, \ldots, w_n) \mapsto (w_1\#^{e_1}, w_2\#^{e_2}, \ldots, w_n\#^{e_n}) \\ \text{with } e_i = (\max_{1\le j\le n} |w_j|) - |w_i| \text{ for } i = 1, \ldots, n.\end{array} \tag{1}$$

This transformation can be viewed as an homogenization and we denote the element thus obtained by $\mathcal{H}_n(w)$ or simply $\mathcal{H}(w)$ when n is understood. For example, with $w = (ab, cdab, \varepsilon, bab)$ we get $\mathcal{H}(w) = (ab\#\#, cdab, \#\#\#\#, bab\#)$. We extend the above notation to all subsets of $(\Sigma^*)^n$: $\mathcal{H}(R) = \{\mathcal{H}(w) \mid w \in R\}$. In particular, the set $\mathcal{H}((\Sigma^*)^n)$ is a subset of the free monoid generated by $(\Sigma\cup\{\#\})\times\cdots\times(\Sigma\cup\{\#\})$. Call *support* of an element $(a_1, \ldots, a_n) \in (\Sigma\cup\{\#\})\times\cdots\times(\Sigma\cup\{\#\})$, the set $\{i \in \{1, \ldots, n\} : a_i \neq \#\}$. An element of the free monoid generated by $(\Sigma\cup\{\#\})\times\cdots\times(\Sigma\cup\{\#\})$ is in $\mathcal{H}((\Sigma^*)^n)$ if and only if it is the concatenation of generators with nonempty nonincreasing supports with respect to the inclusion relation.

In case Σ is finite, the relation $R \subseteq (\Sigma^*)^n$ is *synchronous* if $\mathcal{H}(R)$ is a recognizable subset of the free monoid $((\Sigma \cup \{\#\}) \times \cdots \times (\Sigma \cup \{\#\}))^*$, which is equivalent to saying, according to Kleene theorem, that there exists a finite automaton over the alphabet $((\Sigma \cup \{\#\}) \times \cdots \times (\Sigma \cup \{\#\})$ which recognizes $\mathcal{H}(R)$. The next paragraph is a discussion on how to extend these notions to infinite alphabets.

2.2. Finite automata over infinite alphabets

The obvious extension to an infinite alphabet Σ of the definition of finite automaton keeps the set of states Q finite, and introduces an infinite set of transitions $\Delta \subseteq Q \times \Sigma \times Q$. For $q, r \in Q$, let us call the set $\Delta_{q,r} = \{a \in \Sigma : (q, a, r) \in \Delta\}$ the *label* of the transition from q to r. The transition relation Δ can also be viewed as a function $Q \times Q \to P(\Sigma)$, where $P(\Sigma)$ denotes the power set of Σ, which maps (q, r) to $\Delta_{q,r}$ and that we shall also denote by Δ. Allowing arbitrary labels in $P(\Sigma)$ with Σ infinite is not in the spirit of finite automata. To get a more reasonable notion, let us fix some *finite* Boolean algebra $\mathfrak{A}$ of subsets of Σ, and consider the class of $\mathfrak{A}$-automata obtained by requiring that all labels be in $\mathfrak{A}$. Thus, we consider a finite $\mathfrak{A}$-automaton as a quintuple $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$ where Q is the finite set of states, I and F are the sets of initial and terminal (or final) states and $\Delta : Q \times Q \to \mathfrak{A}$ is the functional representation of the transition relation. We recall that a *run* in $\mathcal{A}$ is a sequence of transitions

$$q_0 \xrightarrow{X_1} q_1 \xrightarrow{X_2} q_2 \cdots q_{i-1} \xrightarrow{X_i} q_i.$$

It is *initial* if q_0 is an initial state and *successful* if furthermore q_i is a final state. Its *label* is the subset concatenation $X_1X_2 \cdots X_i$. The subset of Σ^* *recognized* (or *accepted*) by $\mathcal{A}$ is the union over all successful runs of the concatenation of their labels. Another equivalent definition of an automaton consists of viewing the transition set as a finite subset of triples of the form (q, X, p) where q, p are states and X is a subset of the algebra $\mathfrak{A}$. In that case, several transitions may be associated with the same pair of states (q, r). The conversion from one definition to the other is straightforward, since it consists of splitting a transition or conversely merging transitions.

The main elementary notions and results of finite automata over finite alphabets extend easily to infinite alphabets, and we shall use them without further reference: deterministic $\mathfrak{A}$-automata are defined in the expected way; the classical subset construction extends with no problem implying thus that the family of subsets of Σ^* accepted by finite $\mathfrak{A}$-automata is a Boolean subalgebra of $P(\Sigma^*)$; there exists a minimal automaton which is unique up to isomorphism and which is equivalent to a given $\mathfrak{A}$-automaton and this automaton is also an $\mathfrak{A}$-automaton; recognizable subsets are exactly the subsets which are unions of classes in a right invariant congruence of Σ^*, etc ....

2.3. EES automata and EES relations over possibly infinite alphabets

In order to speak of automata recognizing n-tuples of words over the possibly infinite alphabet Σ, we proceed as in the case of finite alphabets. Indeed, let # be the padding symbol as in paragraph 2.1 and consider the free monoid generated by the subset $(\Sigma \cup \{\#\})^n$.

Definition 2.1. Let Σ be a finite or infinite alphabet.
1. An n-tape EES automaton or simply an EES automaton when n is understood, is an $\mathfrak{A}$-automaton where $\mathfrak{A}$ is a finite Boolean subalgebra of $P((\Sigma \cup \{\#\})^n)$.

2. An n-ary relation $R \subseteq (\Sigma^*)^n$ is EES if $\mathcal{H}_n(R)$ is recognized by some EES automaton.

We shall somehow improperly say that R – instead of $\mathcal{H}_n(R)$ – is recognized by some EES automaton. For future use, we mention the following folklore closure result. Since this concept of folklore is arguable, we briefly sketch the proof which will be probably skipped by most readers.

Proposition 2.2. *The family of EES relations is closed under projections and Cartesian product. The family of n-ary EES relations is closed under Boolean operations.*

Proof. Closure under the Boolean operations follows from the following equalities, where $R, S \subseteq \Sigma^* \times \cdots \times \Sigma^*$ are n-ary EES relations and the trivial observation that $\mathcal{H}(\Sigma^* \times \cdots \times \Sigma^*)$ is itself EES as can be readily verified.

$$\mathcal{H}(R \cup S) = \mathcal{H}(R) \cup \mathcal{H}(S), \qquad \mathcal{H}((\Sigma^* \times \cdots \times \Sigma^*) \setminus R) = \mathcal{H}(\Sigma^* \times \cdots \times \Sigma^*) \setminus \mathcal{H}(R).$$

Now, let $p : (\Sigma^*)^n \to (\Sigma^*)^{n-1}$ be the projection which maps $(u_1, \dots, u_n)$ onto $(u_1, \dots, u_{n-1})$. Consider the morphism $\pi : ((\Sigma \cup \{\#\})^n)^* \to ((\Sigma \cup \{\#\})^{n-1})^*$ between free monoids associated to the projection $(\Sigma \cup \{\#\})^n \to (\Sigma \cup \{\#\})^{n-1}$ between their sets of generators. I.e., π maps the generator $(a_1, \dots, a_n)$ to the generator $(a_1, \dots, a_{n-1})$. Also, denote by g_k, for an arbitrary k, the morphism of $((\Sigma \cup \{\#\})^k)^*$ into itself which erases $(\overbrace{\#, \dots, \#}^{k \text{ times}})$ and leaves all other elements invariant. Then we have

$$\mathcal{H}_{n-1}(p(R)) = g_{n-1}(\pi(\mathcal{H}_n(R))).$$

Concerning the direct product, we consider an n-ary EES relation $R \subseteq (\Sigma^*)^n$ and an m-ary EES relation $S \subseteq (\Sigma^*)^m$. Denote by $\pi_1 : ((\Sigma \cup \{\#\})^{n+m})^* \to ((\Sigma \cup \{\#\})^n)^*$ and $\pi_2 : ((\Sigma \cup \{\#\})^{n+m})^* \to ((\Sigma \cup \{\#\})^m)^*$ the morphisms between free monoids which map the generator $(a_1, \ldots, a_n, a_{n+1}, \ldots, a_{n+m})$ to $(a_1, \ldots, a_n)$ and $(a_{n+1}, \ldots, a_{n+m})$ respectively. Then we have

$$\mathcal{H}(R \times S) = g_{n+m}\left(\pi_1^{-1}\left(\mathcal{H}(R)(\overbrace{\#, \ldots, \#}^{n \text{ times}})^*\right) \cap \pi_2^{-1}\left(\mathcal{H}(S)(\overbrace{\#, \ldots, \#}^{m \text{ times}})^*\right)\right).$$

The result follows from the previous closure properties. □

2.4. The algebras $\mathfrak{F}^n_{\Sigma_0}$ of finitary labels over a possibly infinite alphabet

To tame an excess of generality for the transitions defined by EES automata, and to restrict them to simple effective subsets, we consider the collection of relations on $\Sigma \cup \{\#\}$ which are definable in the structure

$$\mathcal{S}^{\Sigma_0 \cup \{\#\}}_{\Sigma \cup \{\#\}} = \langle \Sigma \cup \{\#\}; =, (a)_{a \in \Sigma_0 \cup \{\#\}} \rangle \tag{2}$$

for some finite subalphabet $\Sigma_0 \subseteq \Sigma$ where # is some fixed symbol outside Σ. We shall see that this amounts precisely to the suggestion of Eilenberg et al. In other words, we shall not develop the general theory of n-tape $\mathfrak{A}$-automata over infinite alphabets and EES relations. We shall merely use this general notion in the statements of Theorems 3.4 and 3.12. Actually, in the vein of this paper, Bès [4] worked out more general notions of finite automata over an infinite alphabet related to richer logics on the alphabet. Let Θ be a quantifier-free formula constructed with n free variables $x_1, \ldots, x_n$, the equality symbol and constant symbols associated with all elements of $\Sigma_0 \cup \{\#\}$. We shall denote by $[\![\Theta]\!]$ the subset of $(\Sigma \cup \{\#\})^n$ defined by Θ in the structure $\mathcal{S}^{\Sigma_0 \cup \{\#\}}_{\Sigma \cup \{\#\}}$, i.e,

$$[\![\Theta]\!] = \{(a_1, \ldots, a_n) \in (\Sigma \cup \{\#\})^n \mid \Theta(a_1, \ldots, a_n)\}. \tag{3}$$

Definition 2.3. The *Boolean algebra of labels* $\mathfrak{F}^n_{\Sigma_0}$ associated to Σ_0 is the trace on $(\Sigma \cup \{\#\})^n \setminus \{\#\}^n$ of the Boolean algebra of n-ary relations on $\Sigma \cup \{\#\}$ which are quantifier-free definable in the structure $\mathcal{S}^{\Sigma_0 \cup \{\#\}}_{\Sigma \cup \{\#\}}$ (cf. (2)). In other words, $\mathfrak{F}^n_{\Sigma_0}$ is the collection of all possible $[\![\Theta]\!]$'s which are disjoint from $\{\#\}^n$. In case $\Sigma_0 = \emptyset$, the algebra $\mathfrak{F}^n_{\Sigma_0}$ is called *constant-free*.

The following result is a straightforward application of the disjunctive normal form of formulae.

Proposition 2.4. 1. *Every n-ary relation in* $\mathfrak{F}^n_{\Sigma_0}$ *is definable in the structure* $\mathcal{S}^{\Sigma_0 \cup \{\#\}}_{\Sigma \cup \{\#\}}$ *by a finite disjunction of formulae which are conjunctions* $\Phi^S_{E,D} \wedge \Psi$ *where*

$$\left.\begin{array}{l} \Psi \text{ is a Boolean combination of expressions of the form } x_i = a \text{ with } 1 \le i \le n \text{ and } a \in \Sigma_0 \\ \Phi^S_{E,D} : \bigwedge\limits_{i \notin S} (x_i = \#) \wedge \bigwedge\limits_{i \in S} (x_i \neq \#) \wedge \bigwedge\limits_{(i,j) \in E} (x_i = x_j) \wedge \bigwedge\limits_{(i,j) \in D} (x_i \neq x_j) \end{array}\right\} \tag{4}$$

and where S, E, D satisfy the conditions

i. $\emptyset \neq S \subseteq \{1, \ldots, n\}$,
ii. $E \subseteq S^2$ *is an equivalence relation on S,*
iii. $D \subseteq S^2$ *is a symmetric relation,*
iv. $D \cap E = \emptyset$ *and* $E \circ D \circ E = D$.

2. *Atoms of* $\mathfrak{F}^n_{\Sigma_0}$ *are the* $[\![\Phi^S_{E,D} \wedge \Psi]\!]$*'s where* $E \cup D = S^2$ *and, for* $1 \le i, j \le n$ *and* $a, b \in \Sigma_0$ *such that* $a \neq b$,

$$\begin{array}{ll} (\Psi \vdash x_i = a) \Leftrightarrow (\Psi \not\vdash x_i \neq a) & (i,j) \in E \Rightarrow \Psi \vdash (x_i = a \Leftrightarrow x_j = a) \\ (\Psi \vdash x_i = a) \Rightarrow (\Psi \vdash x_i \neq b) & (i,j) \in D \Rightarrow \Psi \vdash (x_i = a \Rightarrow x_j \neq a). \end{array}$$

Example 2.5. With $n = 5$, the 5-tuple $(a, a, b, b, \#)$ where a and b are two different letters in Σ, satisfies formula (4) with $S = \{1, 2, 3, 4\}$ and $E = (\{1, 2\} \times \{1, 2\}) \cup (\{3, 4\} \times \{3, 4\})$, $D = S^2 - E$, i.e.,

$$(x_5 = \#) \wedge \bigwedge_{1 \le i \le 4} (x_i \neq \#) \wedge (x_1 = x_2) \wedge (x_3 = x_4) \wedge (x_1 \neq x_3) \wedge (x_1 \neq x_4) \wedge (x_2 \neq x_3) \wedge (x_2 \neq x_4).$$

It also satisfies the formula with $S = \{1, 2, 3, 4\}$ and $E = \{(1, 1), (2, 2)\} \cup (\{3, 4\} \times \{3, 4\})$, $D = S^2 \setminus (E \cup (\{1, 2\} \times \{1, 2\})$.

For readability purposes, we shall sometimes omit trivially deducible equations and inequations. E.g., if $x_1 = x_2$ and $x_2 \neq x_3$ holds, we can deduce $x_1 \neq x_3$ and therefore omit it.

2.5. A characterization of the finitary labels over a possibly infinite alphabet

In this paragraph we characterize the algebras of finitary labels as those invariant under all permutations acting as the identity on a fixed finite subset, as suggested in the paper of Eilenberg et al. In order to avoid cumbersome notations with the # symbol, we state the results of this section with alphabets A and A_0 which are to be $\Sigma \cup \{\#\}$ and $\Sigma_0 \cup \{\#\}$ in the applications. Though we still use the term "alphabet", the semantic is irrelevant; in this paragraph, A is an arbitrary finite or infinite set.

We denote by $\mathfrak{S}_{A_0}(A)$ the family of permutations of A which act as the identity on A_0. We also denote by $\sim_{n,A_0}$ the equivalence on A^n such that

$$\begin{aligned}(x_1, \ldots, x_n) \sim_{n,A_0} (y_1, \ldots, y_n) \ \Leftrightarrow\ & \bigwedge_{1\le i<j\le n} x_i = x_j \Leftrightarrow y_i = y_j \\ & \wedge \bigwedge_{a\in A_0} x_i = a \Leftrightarrow y_i = a.\end{aligned}$$

The following proposition is straightforward.

Proposition 2.6. *Let A be a finite or infinite alphabet and $x_1, \ldots, x_n, y_1, \ldots, y_n \in A$. Then $(x_1, \ldots, x_n) \sim_{n,A_0} (y_1, \ldots, y_n)$ if and only if there exists a permutations in $\mathfrak{S}_{A_0}(A)$ which exchanges $(x_1, \ldots, x_n)$ and $(y_1, \ldots, y_n)$.*

We shall need Proposition 2.6 and Condition ii of the next result for the proof of Theorem 3.4.

Theorem 2.7. *Let A be a finite or infinite set and $\Gamma \subseteq A^n$. The following conditions are equivalent for all finite subsets $A_0 \subseteq A$:*

i. *Γ is invariant under all permutations in $\mathfrak{S}_{A_0}(A)$*
ii. *The family $\{\pi(\Gamma) \mid \pi \in \mathfrak{S}_{A_0}(A)\}$*
   - *is finite if A is infinite,*
   - *has at most $\max(1, |A \setminus A_0| - n)$ elements if A is finite*
iii. *Γ is quantifier-free definable in the structure $\langle A; =, (a)_{a\in A_0}\rangle$*
iv. *Γ is first-order definable in the structure $\langle A; =, (a)_{a\in A_0}\rangle$*
v. *Γ is saturated under the equivalence $\sim_{n,A_0}$ (i.e., Γ is a union of classes).*

Proof. Implications i $\Rightarrow$ ii and iii $\Rightarrow$ iv $\Rightarrow$ i are trivial. Also, Proposition 2.6 yields iii $\Leftrightarrow$ v. We prove ii $\Rightarrow$ iii by induction on $n \ge 1$.
We suppose A is finite, the proof in case A is infinite being similar. We also suppose $|A \setminus A_0| \ge 2$ otherwise every relation on A is definable in the structure $\langle A; =, (a)_{a\in A_0}\rangle$ and iii is trivial.

Basic step $n = 1$. We prove the following more precise property:

$$\text{either } \Gamma \subseteq A_0 \text{ or } \Gamma \supseteq A \setminus A_0. \qquad (*)$$

Let $k = |\Gamma \setminus A_0|$ and $\ell = |A \setminus A_0|$. If $(*)$ were false, then we would have $0 < k < \ell$ implying $\binom{\ell}{k} > \ell$. Now, any two subsets of $A \setminus A_0$ with cardinality k can be exchanged by a permutation in $\mathfrak{S}_{A_0}(A)$. In particular, $\{\pi(\Gamma) \mid \pi \in \mathfrak{S}_{A_0}(A)\}$ has at least $\binom{\ell}{k} > \ell - 1$ elements. Which contradicts ii.

Inductive step: from n to $n + 1$. Let $\Gamma \subseteq A^{n+1}$ satisfy ii. For $a_1, \ldots, a_n \in A$, set $\Gamma_{a_1,\ldots,a_n} = \{a \mid (a_1, \ldots, a_n, a) \in \Gamma\}$. Let also $A_0^{a_1,\ldots,a_n} = \{a_1, \ldots, a_n\} \cup A_0$. Then

$$\begin{aligned}\{\pi(\Gamma_{a_1,\ldots,a_n}) \mid \pi \in \mathfrak{S}_{A_0^{a_1,\ldots,a_n}}(A)\} &= \{\pi(\Gamma)_{a_1,\ldots,a_n} \mid \pi \in \mathfrak{S}_{A_0^{a_1,\ldots,a_n}}(A)\} \\ &\subseteq \{\pi(\Gamma)_{a_1,\ldots,a_n} \mid \pi \in \mathfrak{S}_{A_0}(A)\}.\end{aligned}$$

Since Γ satisfies condition ii, we have

$$\begin{aligned}|\{\pi(\Gamma_{a_1,\ldots,a_n}) \mid \pi \in \mathfrak{S}_{A_0^{a_1,\ldots,a_n}}(A)\}| &\le |\{\pi(\Gamma) \mid \pi \in \mathfrak{S}_{A_0}(A)\}| \\ &\le \max(1, |A \setminus A_0| - (n + 1)) \\ &\le \max(1, |A \setminus A_0^{a_1,\ldots,a_n}| - 1).\end{aligned}$$

Using the above basic step, this last inequality allows us to use $(*)$ for the set $\Gamma_{a_1,\ldots,a_n}$ with $A_0^{a_1,\ldots,a_n}$ in place of A_0. Thus, $\Gamma_{a_1,\ldots,a_n}$ is included in $A_0^{a_1,\ldots,a_n}$ or contains $A \setminus A_0^{a_1,\ldots,a_n}$. Let us introduce the following relations where $X \subseteq A_0$ and $I \subseteq \{1, \ldots, n\}$:

$$\begin{aligned}\mu_{X,I} &= \{(a_1, \ldots, a_n) \in A^n \mid \Gamma_{a_1,\ldots,a_n} = X \cup \{a_i \mid i \in I\}\} \\ \nu_{X,I} &= \{(a_1, \ldots, a_n) \in A^n \mid \Gamma_{a_1,\ldots,a_n} = X \cup \{a_i \mid i \in I\} \cup A \setminus A_0^{a_1,\ldots,a_n}\} \\ \Gamma^{\mu}_{X,I} &= \{(a_1, \ldots, a_n, a_{n+1}) \mid (a_1, \ldots, a_n) \in \mu_{X,I} \wedge a_{n+1} \in X \cup \{a_i \mid i \in I\}\} \\ \Gamma^{\nu}_{X,I} &= \{(a_1, \ldots, a_n, a_{n+1}) \mid (a_1, \ldots, a_n) \in \nu_{A,I} \wedge\ a_{n+1} \in X \cup \{a_i \mid i \in I\} \cup (A \setminus A_0^{a_1,\ldots,a_n})\}).\end{aligned}$$

For any X, I and $\pi \in \mathfrak{S}_{A_0}(A)$ we have

$$\begin{aligned}\pi(\mu_{X,I}) &= \{(\pi(a_1),\ldots,\pi(a_n)) \in A^n \mid (\pi(\Gamma))_{\pi(a_1),\ldots,\pi(a_n)} = X \cup \{\pi(a_i) \mid i \in I\}\} \\ &= \{(b_1,\ldots,b_n) \in A^n \mid \pi(\Gamma)_{b_1,\ldots,b_n} = X \cup \{b_i \mid i \in I\}\}.\end{aligned}$$

Therefore $|\{\pi(\mu_{X,I}) \mid \pi \in \mathfrak{S}_{A_0}(A)\}| \leq |\{\pi(\Gamma) \mid \pi \in \mathfrak{S}_{A_0}(A)\}|$. In particular, this implies condition ii for $\mu_{X,I}$. A similar property holds with $\nu_{X,I}$. Using the induction hypothesis, $\mu_{X,I}$ and $\nu_{X,I}$ are quantifier-free definable in $\langle A; =, (a)_{a\in A_0}\rangle$ and so are $\Gamma^{\mu}_{X,I}$ and $\Gamma^{\nu}_{X,I}$. Since $\Gamma = \bigcup_{X,I}(\Gamma^{\mu}_{X,I} \cup \Gamma^{\nu}_{X,I})$, this leads to a quantifier-free definition of Γ in $\langle A; =, (a)_{a\in A_0}\rangle$. $\square$

The rest of this paragraph investigates definability with different finite subsets $A_0 \subseteq A$. The results will be used in Section 3.1. Let us introduce two convenient notations in the vein of (2) : for $B \subseteq A$,

$$\mathcal{S}_A = \langle A; =, (a)_{a\in A}\rangle \qquad \mathcal{S}_A^B = \langle A; =, (a)_{a\in B}\rangle.$$

Proposition 2.8. *Let A_1, A_2 be two finite subsets of A, such that $A_1 \cup A_2 \neq A$. Then a relation $\Gamma \subseteq A^n$ is definable in $\mathcal{S}_A^{A_1}$ and in $\mathcal{S}_A^{A_2}$ if and only if it is definable in $\mathcal{S}_A^{A_1\cap A_2}$. In particular, if A is infinite, then for every relation Γ definable in $\mathcal{S}_A$ there exists a smallest finite subset $A_0 \subseteq A$ such that Γ is definable in $\mathcal{S}_A^{A_0}$.*

Observe that the condition $A_1 \cup A_2 \neq A$ always holds if A is infinite and that the statement fails when $A_1 \cup A_2 = A$. For instance, if $A = \{a_i, b_i \mid i = 1,\ldots,k\}$ and $A_i = A \setminus \{a_i, b_i\}$ and $R = \{(a_i, b_i), (b_i, a_i) \mid i = 1,\ldots,k\}$ then the A_i's are the minimal subalphabets $B \subseteq A$ such that R is definable in $\mathcal{S}_A^B$.

Proof. Using Theorem 2.7, we are reduced to prove that if Γ is invariant under all permutations in $\mathfrak{S}_{A_1}(A) \cup \mathfrak{S}_{A_2}(A)$ then it is also invariant under all permutations in $\mathfrak{S}_{A_1\cap A_2}(A)$. In order to simplify notations, we identify $A_1 \cup A_2$ with a set of positive integers and we assume

$$\begin{aligned} A_1 \setminus A_2 &= \{1,\ldots,p\} \\ A_2 \setminus A_1 &= \{p+1,\ldots,p+m\} \end{aligned} \qquad A_1 \cap A_2 = \{p+m+1,\ldots,p+m+q\}.$$

We show that Γ is invariant under all permutations π fixing each element of $(A_1 \cap A_2) \cup \{1,\ldots,p-1\} = A_1 \setminus \{p\}$, which proves the statement by induction on the cardinality of $A_1 \setminus A_2$.

Since π leaves each element in $A_1 \cap A_2$ invariant, we have $\pi(p) \notin A_1 \cap A_2$.

Consider first the case where $\pi(p) \notin A_2 \setminus A_1$. Then both $p, \pi(p)$ are outside A_2. Let α be the transposition exchanging $\pi(p)$ and p. Then $\alpha = \alpha^{-1} \in \mathfrak{S}_{A_2}(A)$, hence $\alpha^{-1}(\Gamma) = \Gamma$. Also, since π leaves each element in $\{1,\ldots,p-1\} \cup (A_1 \cap A_2)$ invariant, $\alpha\pi$ leaves invariant each element in $\{1,\ldots,p\} \cup (A_1 \cap A_2) = A_1$ invariant, hence $\alpha\pi \in \mathfrak{S}_{A_1}(A)$ and $\alpha\pi(\Gamma) = \Gamma$. Thus $\pi(\Gamma) = \alpha^{-1}\alpha\pi(\Gamma) = \Gamma$.

Now, if $\pi(p) \in \{p+1,\ldots,p+m\} = A_2 \setminus A_1$, consider the transposition β exchanging $\pi(p)$ with some element outside $A_1 \cup A_2$. It leaves each element in $\{1,\ldots,p\} \cup (A_1 \cap A_2) = A_1$ invariant, hence $\beta = \beta^{-1} \in \mathfrak{S}_{A_1}(A)$ and $\beta^{-1}(\Gamma) = \Gamma$. Furthermore we have $\beta\pi(p) \notin \{p+1,\ldots,p+m\}$ and $\beta\pi$ fixes each element in $\{1,\ldots,p-1\}$. Because of the previous discussion we have $\beta\pi(\Gamma) = \Gamma$. Finally, we obtain $\pi(\Gamma) = \beta^{-1}\beta\pi(\Gamma) = \Gamma$. $\square$

Proposition 2.9. *Let A be finite or infinite alphabet. There is an algorithm which, given two finite subsets A_1 and A_2 of A and a definition in $\mathcal{S}_A^{A_1}$ of a relation Γ, decides whether or not Γ is definable in the structure $\mathcal{S}_A^{A_2}$.*

Proof. To check whether $\Gamma \subseteq A^n$ is definable in $\mathcal{S}_A^{A_2}$, we use condition iv of Theorem 2.7. Let $\psi(x_1,\ldots,x_n,y_1,\ldots,y_n)$ be the formula

$$\bigwedge_{1\leq i\leq n}\bigwedge_{a\in A_2}(x_i = a \Leftrightarrow y_i = a) \;\wedge \bigwedge_{1\leq i<j\leq n}(x_i = x_j \Leftrightarrow y_i = y_j)$$

which defines the equivalence $\sim_{n,A_2}$ in $\mathcal{S}_A^{A_2}$. Given a formula $\phi(x_1,\ldots,x_n)$ using constants in A_1 which defines a relation Γ in $\mathcal{S}_A^{A_1}$, we know that Γ is definable in $\mathcal{S}_A^{A_2}$ if and only if it is $\sim_{n,A_2}$-saturated. This is expressible in $\mathcal{S}_A^{A_1\cup A_2}$ as follows:

$$\forall x_1 \ldots \forall x_n\, \forall y_1 \ldots \forall y_n \quad (\psi(x_1,\ldots,x_n,y_1,\ldots,y_n) \Rightarrow (\phi(x_1,\ldots,x_n) \Leftrightarrow \phi(y_1,\ldots,y_n))).$$

Since the structure $\mathcal{S}_A$ (with all possible constants) admits effective quantifier elimination, the above formula can be effectively tested. $\square$

3. Synchronous and oblivious synchronous relations

The families of automata mentioned in the table of Fig. 1 are specified via the families of labels of their transitions. This section is devoted to their investigation. These families make sense, and are interesting no matter whether the alphabet Σ is finite or infinite. Indeed, for a finite alphabet Σ (and a subalphabet Σ_0 satisfying $|\Sigma \setminus \Sigma_0| \geq 2$), the families of Σ_0-synchronous n-ary relations and that of oblivious Σ_0-synchronous n-ary relations are Boolean algebras which lie strictly between the class of recognizable relations and that of synchronous (in the usual sense) relations.

Though some key results hold only in case Σ is infinite, most of them, especially in Section 4 hold in both cases and lead to a refinement of the main theorem of Eilenberg et al. [9].

3.1. Synchronous automata and synchronous relations

Definition 3.1. Let Σ be a finite or infinite alphabet.
1. Let Σ_0 be a finite subalphabet of Σ. An automaton $\mathcal{A}$ is Σ_0-*synchronous* if its labels lie in the finite Boolean algebra $\mathfrak{F}^n_{\Sigma_0}$, i.e., $\mathcal{A}$ is an n-tape $\mathfrak{F}^n_{\Sigma_0}$-automaton in the sense of Section 2.3. Furthermore, the automaton is *constant-free* synchronous whenever $\Sigma_0 = \emptyset$ holds.

For convenience, we shall often split a transition $q \xrightarrow{\bigcup_{1\le i\le m} [\![\Phi^{S_i}_{E_i,D_i} \wedge \Psi_i]\!]} r$ into m transitions $q \xrightarrow{[\![\Phi^{S_i}_{E_i,D_i} \wedge \Psi_i]\!]} r$, $i = 1, \ldots, m$.

2. A relation $R \subseteq (\Sigma^*)^n$ is Σ_0-*synchronous* if there exists an n-tape Σ_0-synchronous automaton such that $\mathcal{H}(R)$ (cf. Section 2.1) is the union of the labels of all successful runs. Constant-free synchronous relations are defined accordingly.

3. An automaton or a relation is *synchronous* if it is Σ_0-synchronous for some finite subalphabet Σ_0 of Σ.

Of course, if Σ is finite then Σ-synchronous means synchronous in the usual sense. However, for $\Sigma_0 \subsetneq \Sigma$, Σ_0-synchronous relations constitute a proper subclass of usual synchronous relations.

Let's introduce one more notion.

Definition 3.2. We denote by $\equiv^{\text{sync}}_{n,\Sigma_0}$ the equivalence relation on n-tuples of words in Σ^* such that $(u_1, \ldots, u_n) \equiv^{\text{sync}}_{n,\Sigma_0} (v_1, \ldots, v_n)$ if the following conditions hold: for $1 \le i \le n$ and $1 \le j < k \le n$,

1. $|u_i| = |v_i|$
2. for $\ell \le |u_i|$, if $u_i[\ell]$ or $v_i[\ell]$ is in Σ_0 then $u_i[\ell] = v_i[\ell]$
3. for $1 \le \ell \le \min\{|u_j|, |u_k|\}$, $u_j[\ell] = u_k[\ell]$ if and only if $v_j[\ell] = v_k[\ell]$.

The following result is straightforward.

Proposition 3.3. $(u_1, \ldots, u_n) \equiv^{sync}_{n,\Sigma_0} (v_1, \ldots, v_n)$ *if and only if for every* $\ell \le \max\{|u_i| : i = 1, \ldots, n\}$, *we have* $\mathcal{H}(u_1, \ldots, u_n)[\ell] \sim_{n,\Sigma_0\cup\{\#\}} \mathcal{H}(v_1, \ldots, v_n)[\ell]$ *(where* $\sim_{n,\Sigma_0\cup\{\#\}}$ *is the equivalence defined in Section 2.5).*

In the case of an infinite alphabet, the following theorem justifies the suggestion of Eilenberg et al. to consider relations invariant under all permutations acting as the identity over a finite subset of Σ. As for the case of a finite alphabet, one has to consider level-by-level permutations, i.e., infinite sequences of permutations $\boldsymbol{\pi} = (\pi_k)_{k\ge1}$ which operate on words by substituting $\pi_k(a_k)$ for the k-th letter a_k

$$\boldsymbol{\pi}(a_1 \cdots a_r) = \pi_1(a_1) \cdots \pi_r(a_r).$$

Theorem 3.4. *Let Σ be a finite or infinite alphabet, let Σ_0 be a finite subalphabet of Σ and let $R \subseteq (\Sigma^*)^n$. The following conditions are equivalent:*

i. *R is Σ_0-synchronous*
ii. *R is an EES relation which is $\equiv^{sync}_{n,\Sigma_0}$-saturated*
iii. *R is the $\equiv^{sync}_{n,\Sigma_0}$-saturation of an EES relation*
iv. *R is an EES relation which is invariant under all level-by-level permutations of Σ which, at every level, act as the identity on Σ_0.*

In case Σ is infinite, one can add a fifth equivalent condition:

v. *R is an EES relation which is invariant under all permutations of Σ in $\mathfrak{S}_{\Sigma_0}(\Sigma)$ (i.e., those which act as the identity on Σ_0).*

The implication v $\Rightarrow$ i fails when Σ is finite and $|\Sigma \setminus \Sigma_0| \ge 2$ (and is trivial if $|\Sigma \setminus \Sigma_0| \le 1$). For instance, $R = \{aa \mid a \in \Sigma \setminus \Sigma_0\}$ is EES (even $(\Sigma \setminus \Sigma_0)$-synchronous) and invariant under all permutations fixing each element in Σ_0 but is not Σ_0-synchronous.

Proof. i $\Rightarrow$ ii. Let $\mathcal{A}$ be a Σ_0-synchronous automaton recognizing R. Using Theorem 2.7, we know that the labels of transitions of $\mathcal{A}$ are $\sim_{n,\Sigma_0\cup\{\#\}}$-saturated. Using Proposition 3.3, we deduce that the labels of runs of $\mathcal{A}$ are $\equiv^{\text{sync}}_{n,\Sigma_0}$-saturated. Hence R is $\equiv^{\text{sync}}_{n,\Sigma_0}$-saturated.

Implication ii $\Rightarrow$ iii is trivial. Let us prove iii $\Rightarrow$ i. Suppose R is the $\equiv^{\text{sync}}_{n,\Sigma_0}$-saturation of an EES relation S recognized by the EES automaton $\mathcal{A}$. Let $\mathcal{B}$ be obtained by saturating the labels of $\mathcal{A}$ for $\sim_{n,\Sigma_0\cup\{\#\}}$. Then $\mathcal{B}$ is a Σ_0-synchronous automaton. Obviously, $\mathcal{B}$ recognizes all elements of S hence also all elements of its saturated R. Using Proposition 3.3, we see that any element of $(\Sigma^*)^n$ recognized by $\mathcal{B}$ is $\equiv^{\text{sync}}_{n,\Sigma_0}$-equivalent to some element recognized by $\mathcal{A}$, hence is in R. Thus, $\mathcal{B}$ recognizes R.

ii $\Rightarrow$ iv. Observe that if the level-by-level permutation $\boldsymbol{\pi} = (\pi_k)_{k\ge1}$ acts as the identity on Σ_0 then $(\boldsymbol{\pi}(u_1), \ldots, \boldsymbol{\pi}(u_n)) \equiv^{\text{sync}}_{n,\Sigma_0} (u_1, \ldots, u_n)$. Using ii, we obtain $\boldsymbol{\pi}(R) \subseteq R$. Arguing with $\boldsymbol{\pi}^{-1} = (\pi_k^{-1})_{k\ge1}$, we obtain $\boldsymbol{\pi}^{-1}(R) \subseteq R$ and then, applying $\boldsymbol{\pi}$, we obtain $R \subseteq \boldsymbol{\pi}(R)$. Whence $R = \boldsymbol{\pi}(R)$.

iv $\Rightarrow$ i. Straightforward from Propositions 3.3 and 2.6 and Theorem 2.7 (equivalence i $\Leftrightarrow$ v).

The above arguments prove the equivalence of i, ii, iii and iv. We now deal with v.
iv $\Rightarrow$ v is trivial. We prove v $\Rightarrow$ i. This implication requires Σ to be infinite. Consider the minimal deterministic n-tape automaton $\mathcal{D}$ recognizing R. Let $\pi(\mathcal{D})$ be obtained from $\mathcal{D}$ by applying π to the labels. Then $\pi(\mathcal{D})$ recognizes $\pi(R)$ and, due to the uniqueness, it is the minimal deterministic n-tape automaton recognizing $\pi(R)$. Now, R is invariant under all permutations $\pi \in \mathfrak{S}_{\Sigma_0}(\Sigma)$ (i.e., those which are the identity on Σ_0). Thus, if $\pi \in \mathfrak{S}_{\Sigma_0}(\Sigma)$ then $\pi(\mathcal{D})$ and $\mathcal{D}$ are the same automaton up to some renaming of states. This proves that the labels of $\pi(\mathcal{D})$ are among the labels of $\mathcal{D}$. In particular, for every label X of $\mathcal{D}$, the family $\{\pi(X) \mid \pi \in \mathfrak{S}_{\Sigma_0}(\Sigma)\}$ is included in the family of labels of $\mathcal{D}$ hence is finite. Applying Theorem 2.7, we see that every label X of $\mathcal{D}$ is in $\mathfrak{F}^n_{\Sigma_0}$. In other words, $\mathcal{D}$ is a Σ_0-synchronous automaton which recognizes R. $\square$

Proposition 2.8 has an analog with synchronous relations.

Theorem 3.5. *Let Σ_1 and Σ_2 be two finite subsets of Σ such that $\Sigma_1 \cup \Sigma_2 \neq \Sigma$ (which is always the case if Σ is infinite). Then a relation $R \subseteq (\Sigma^*)^n$ is Σ_1 and Σ_2-synchronous if and only if it is $(\Sigma_1 \cap \Sigma_2)$-synchronous.*
In particular, if Σ is infinite then for every synchronous relation R there exists a smallest finite subset $\Sigma_0 \subseteq \Sigma$ such that R is Σ_0-synchronous. Furthermore, this smallest subalphabet Σ_0 can be effectively computed.

Observe that the condition $\Sigma_1 \cup \Sigma_2 \neq \Sigma$ is necessarily satisfied for infinite alphabets. For finite alphabets, the result no longer holds when the inequality fails. Indeed, it suffices to consider the counterexample of Proposition 2.8: a $\mathcal{S}_{\Sigma_0 \cup \{\#\}}$-definable relation in Σ^n is, in particular, a Σ_0-synchronous relation in $(\Sigma^*)^n$.

Proof. The relation is Σ_i-synchronous if and only if the transitions of its minimal automaton are definable in $\langle \Sigma \cup \{\#\}; = (a)_{a \in \Sigma_i \cup \{\#\}} \rangle$. We conclude using Propositions 2.8 and 2.9 with $A = \Sigma \cup \{\#\}$ and $A_i = \Sigma_i \cup \{\#\}$. $\square$

Observe that the the closure properties of EES relations mentioned in Proposition 2.2 are also valid for the family of $\equiv^{\mathrm{sync}}_{n,\Sigma_0}$-saturated relations. Therefore, using condition ii in Theorem 3.4, we can extend these closure properties of EES relations to synchronous relations.

Corollary 3.6. *Let Σ_1 and Σ_2 be two finite subsets of Σ and let R_1 be a Σ_1-synchronous relation and R_2 be a Σ_2-synchronous relation. Let p be the projection defined by $p(w_1, \ldots, w_n) \mapsto (w_{i_1}, \ldots, w_{i_k})$ where n is the arity of R_1 and the i_j's are among $1, \ldots, n$.*
Then $p(R_1)$ and $(\Sigma^*)^n \setminus R_1$ are Σ_1-synchronous and $R_1 \times R_2$ is $(\Sigma_1 \cup \Sigma_2)$-synchronous. If R_1 and R_2 have the same arity then $R_1 \cup R_2$ and $R_1 \cap R_2$ are $(\Sigma_1 \cup \Sigma_2)$-synchronous.
Moreover, all these closure properties are effective in terms of synchronous automata.

Using the decidability of the emptiness problem, we obtain the following corollary.

Corollary 3.7. *Let Σ be finite or infinite alphabet. There is an algorithm which, given two synchronous automata, decides whether or not they recognize the same relation on Σ^*.*

Let us state a last decision property.

Theorem 3.8. *Let Σ be finite or infinite alphabet. There is an algorithm which, given finite subalphabets Σ_0 and Σ_1 of Σ and a Σ_1-synchronous automaton $\mathcal{A}$, decides if the relation R recognized by $\mathcal{A}$ is Σ_0-synchronous.*

Proof. As in the proof of iii $\Rightarrow$ i in Theorem 3.4, from $\mathcal{A}$ we effectively construct an automaton $\mathcal{B}$ which recognizes the $\equiv^{\mathrm{sync}}_{n,\Sigma_0}$-saturation of R. Now, by the equivalence i $\Leftrightarrow$ ii of Theorem 3.4, the relation R is Σ_0-synchronous if and only if the two automata $\mathcal{A}$ and $\mathcal{B}$ recognize the same relation. $\square$

3.2. *Oblivious synchronous automata and oblivious synchronous relations*

Extending the main result of [9] to infinite alphabets requires to introduce a new type of synchronous automata. We call them oblivious because their ability to detect equality of the letters on a given pair of distinct tapes vanishes after the first negative check for that pair.

Before giving a formal definition of our class of automata, we describe intuitively how they work. The idea is to view a computation on an n-tuple $(w_1, \ldots, w_n) \in \mathcal{H}(\Sigma^*)^n$ (cf. Section 2.1) as the following process involving time:

(*) At time t the automaton reads the t-th letters $(w_1[t], \ldots, w_n[t])$ of each component simultaneously.
(**) Equality between a pair of components of an n-tuple may be tested if and only if it was previously true without interruption. After an interruption, the automaton is no longer able to test equality or inequality between these two components at any further step. For example, the automaton may require the first two components to be equal up to the value t, namely $w_1[1] = w_2[1]$, $w_1[2] = w_2[2]$,…, $w_1[t] = w_2[t]$, but if the automaton fails to maintain this requirement at $t + 1$, i.e., if $w_1[t + 1] \neq w_2[t + 1]$, it will no longer be able to test $w_1[t'] = w_2[t']$ for $t' > t + 1$.

With this in mind we turn to the formal definition of an oblivious synchronous automaton $\mathcal{O}$ for a finite or infinite alphabet Σ. It consists of restricting the possible labels of a transition, leaving a given state.

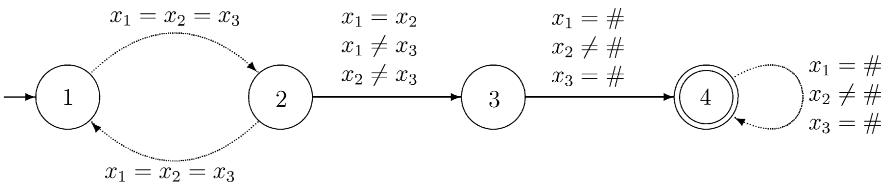

Fig. 2. A constant-free oblivious synchronous automaton.

Definition 3.9 (*Oblivious Synchronous Automaton*). (1) An n-tape Σ_0-synchronous automaton $\mathcal{O}$ (cf. Definition 3.1) is *oblivious* if the following conditions are satisfied.

i. The states are of the form (q, S, E) where
   - q belongs to a finite set Q,
   - $\emptyset \neq S \subseteq \{1, \ldots, n\}$ tells which components are in Σ,
   - E is an equivalence relation on S.

ii. A state is final if its first component belongs to a specific subset $F \subseteq Q$.

iii. Initial states are the triples $(q, \{1, \ldots, n\}, \{1, \ldots, n\} \times \{1, \ldots, n\})$ where the first component belongs to a specific subset $I \subseteq Q$.

iv. The transitions with non empty labels are of the form

$$(q', S', E') \xrightarrow{[\![\Phi]\!]} (q, S, E)$$

where $[\![\Phi]\!]$ and $\Phi \equiv \Phi^S_{E,D} \wedge \Psi$ are as in (3) and (4) (cf. Section 2.4) and Proposition 2.4. Furthermore the following conditions hold

$$S \subseteq S', \qquad E \subseteq E', \qquad E' \cap S^2 = E \cup D. \tag{5}$$

(2) A relation $R \subseteq (\Sigma^*)^n$ is *oblivious* Σ_0*-synchronous* if it is recognized by an oblivious Σ_0-synchronous automaton. Constant-free oblivious synchronous automata and relations correspond to the case $\Sigma_0 = \emptyset$.

(3) An automaton or a relation is *oblivious synchronous* if it is oblivious Σ_0-synchronous for some finite subalphabet Σ_0 of Σ.

Of course, if Σ is finite then oblivious Σ-synchronous means synchronous in the usual sense. However, for $\Sigma_0 \subsetneq \Sigma$, none of the following implications can be reversed:

$$\textit{oblivious}\,\Sigma_0 - \textit{synchronous} \Rightarrow \Sigma_0 - \textit{synchronous} \Rightarrow \textit{usualsynchronous}.$$

We would like to draw the attention to the touchy point of the definition, since it is the crux of our characterization. Inclusions $S \subseteq S'$, $E \subseteq E'$ and $D \subseteq E'$ amount to inclusion $E' \cap S^2 \supseteq E \cup D$ and convey the "only if" part of condition $(**)$ (cf. top of this §). The converse inclusion $E' \cap S^2 \subseteq E \cup D$ conveys the "if" part. Indeed, if the variables x_i and x_j are maintained equal, i.e., if $(i, j) \in E'$ then we may impose to keep them equal or to make them non-equal, but if $(i, j) \notin E'$, then there is no way we can control their equality or inequality, except via equality or inequality with some constant in Σ_0.

Of course, if Σ is finite then Σ-synchronous means synchronous in the usual sense. Therefore, for $\Sigma_0 \subsetneq \Sigma$, Σ_0-synchronous relations constitute a proper subclass of usual synchronous relations.

3.3. *Examples of synchronous and oblivious synchronous relations*

The automaton in Fig. 2 recognizes the constant-free oblivious synchronous relation

$$R = \{(ua, uav, ub) \mid u, v \in \Sigma^*, |v| \geq 1, |u| = 1 \bmod 2, a, b \in \Sigma, a \neq b\}.$$

The second and third components in the states (i.e., the S and E in the expression (q, S, E)) are defined as follows

$$\begin{array}{rlrl} S_1 = S_2 &= \{1, 2, 3\} & E_1 = E_2 &= \{1, 2, 3\}^2 \\ S_3 &= \{1, 2, 3\} & E_3 &= \{1, 2\}^2 \cup \{3\}^2, \\ S_4 &= \{2\} & E_4 &= \{2\}^2. \end{array}$$

Observe that from state 2 to state 3 the label contains the condition $x_1 \neq x_3$, which is allowed because the transition leaves state 2 where x_1 and x_3 are supposed to be equal. The same condition could not possibly be part of a label of a transition leaving state 3 because from that state on, x_1 and x_3 can no longer be compared. Though not explicitly written, the subformulae $\Phi^S_{E,D}$ and Ψ are understood from the context.

Example 3.10. The binary relation `EqLenEqLast` (cf. Section 1.1, Problem 1) is recognized by the constant-free synchronous automaton in Fig. 3, where $\mathtt{Diag} = \{(a, a) \in \Sigma \times \Sigma : a \in \Sigma\} = [\![x_1 = x_2]\!]$.

The unary relation $\mathtt{Last}_a$ is recognized by the $\{a\}$-synchronous automaton in Fig. 4 where the two labels of transitions are $\Sigma \setminus \{a\} = [\![x_1 \neq \# \wedge x_1 \neq a]\!]$ and $\{a\} = [\![x_1 = a]\!]$.

The unary relation $\mathtt{mod}_{k,\ell}$ (cf. Section 1.2, Point 5) is recognized by the constant-free synchronous automaton in Fig. 5 where all labels are $\Sigma = [\![x_1 = x_1]\!]$.

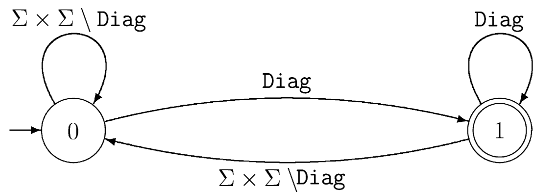

Fig. 3. A constant-free synchronous automaton for `EqLenEqLast`.

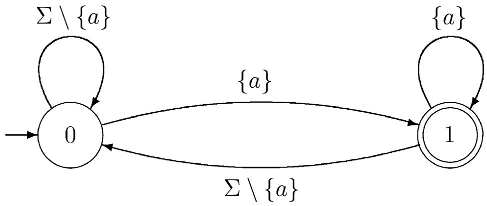

Fig. 4. An $\{a\}$-synchronous automaton for the predicate $\mathtt{Last}_a$.

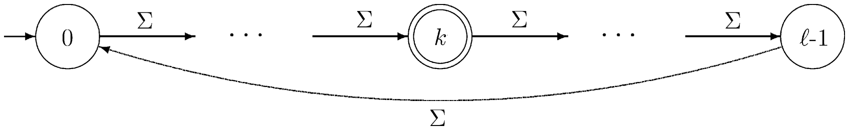

Fig. 5. A constant-free oblivious synchronous automaton for the relation $\mathtt{mod}_{k,\ell}$.

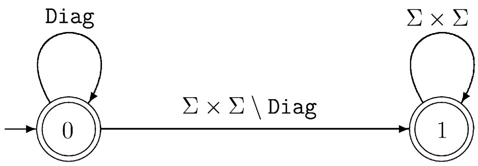

Fig. 6. A constant-free oblivious synchronous automaton for the relation `EqLen`.

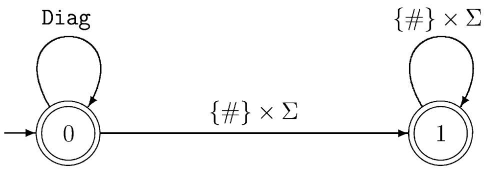

Fig. 7. A constant-free oblivious synchronous automaton for the relation `Pref`.

The relation `EqLen` is recognized by the constant-free oblivious synchronous automaton in Fig. 6. Denoting by (S_0, E_0) and (S_1, E_1) the second and third components in states 0 and 1, we have $S_0 = S_1 = \{1, 2\}$ $E_0 = \{1, 2\} \times \{1, 2\}$ and $E_1 = \{(1, 1), (2, 2)\}$. Due to condition iii about initial states in Definition 3.9, this relation is recognizable by no oblivious automaton with a unique state.

The relation `Pref` is recognized by the constant-free oblivious synchronous automaton in Fig. 7 where $S_0 = \{1, 2\}$, $S_1 = \{2\}$, $E_0 = \{1, 2\} \times \{1, 2\}$ and $E_1 = \{(2, 2)\}$.

3.4. Relationship between synchronous and oblivious synchronous

The general problem is the following: given a Σ_0-synchronous automaton, is it recursively decidable whether or not it is oblivious Σ_0-synchronous? or oblivious Σ_1-synchronous for some given finite subset Σ_1? Our proof relies on the following notion which is the oblivious analog of that of Definition 3.2.

Definition 3.11. We denote by $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$ the equivalence relation on n-tuples of words in Σ^* such that $(u_1, \ldots, u_n) \equiv^{\mathrm{obl}}_{n,\Sigma_0} (v_1, \ldots, v_n)$ if the following conditions hold: for $1 \le i \le n$ and $1 \le j < k \le n$,

1. $|u_i| = |v_i|$
2. for $\ell \le |u_i|$, if $u_i[\ell]$ or $v_i[\ell]$ is in Σ_0 then $u_i[\ell] = v_i[\ell]$
3. for $1 \le \ell \le \min\{|u_j|, |u_k|\}$, $u_j \upharpoonright \ell = u_k \upharpoonright \ell$ if and only if $v_j \upharpoonright \ell = v_k \upharpoonright \ell$.

The next result summarizes the connections between the notions of being synchronous, oblivious synchronous and saturated.

Theorem 3.12. *Let Σ be a finite or infinite alphabet, let Σ_0 be a finite subalphabet of Σ and let $R \subseteq (\Sigma^*)^n$. The following conditions are equivalent.*

i. *R is oblivious Σ_0-synchronous*
ii. *R is an EES relation which is $\equiv^{obl}_{n,\Sigma_0}$-saturated*
iii. *R is the $\equiv^{obl}_{n,\Sigma_0}$-saturation of an EES relation.*

For all finite subalphabets Σ_1, Σ_2 such that $\Sigma_1 \cap \Sigma_2 = \Sigma_0$ and $|\Sigma \setminus (\Sigma_1 \cup \Sigma_2)| \geq n$ holds, there is a fourth equivalent condition:

iv. *R is Σ_2-synchronous and oblivious Σ_1-synchronous.*

The implication iv $\Rightarrow$ i fails when $|\Sigma \setminus (\Sigma_1 \cup \Sigma_2)| < n$. For instance, if $|\Sigma \setminus \Sigma_1| \leq 1$, then any synchronous relation (in the usual sense) is oblivious Σ_1-synchronous. Whereas, if $|\Sigma \setminus \Sigma_2| \geq 2$ then there are Σ_2-synchronous relations which are not oblivious Σ_2-synchronous.

Proof. i $\Rightarrow$ ii. A routine argument shows that the label of a run of an oblivious automaton is $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturated.

The implication ii $\Rightarrow$ iii is trivial. Let us prove iii $\Rightarrow$ i. The idea is as in the proof of the similar implication in Theorem 3.4: we consider an EES automaton $\mathcal{A}$, which recognizes a relation T and transform it into an oblivious Σ_0-synchronous automaton $\mathcal{O}$ which recognizes the $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturation R of T. However, the construction is a little more technical. First, observe that the equivalence $\equiv^{\mathrm{sync}}_{n,\Sigma_0}$ refines $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$. Therefore R is also the $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturation of the $\equiv^{\mathrm{sync}}_{n,\Sigma_0}$-saturation U of T. Using Theorem 3.4, we know that U is Σ_0-synchronous. Thus, we are reduced to the case where T is itself Σ_0-synchronous. Let $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$ be a Σ_0-synchronous automaton which recognizes T. After possibly splitting the labels, we may assume that all labels $[\![\Phi^S_{E,D} \wedge \Psi]\!]$ of the transitions are atoms of the algebra $\mathfrak{F}^n_{\Sigma_0}$, cf. Proposition 2.4. Define the oblivious Σ_0-synchronous automaton $\mathcal{O} = (\widetilde{Q}, \Sigma, \widetilde{\Delta}, \widetilde{I}, \widetilde{F})$ as follows:

(a) $\widetilde{Q}$ is the set of triples (q, S, E) where $q \in Q$ and $\emptyset \neq S \subseteq \{1, \ldots, n\}$ and E is an equivalence relation on S.
(b) $\widetilde{I}$ is the set of triples $(q, \{1, \ldots, n\}, \{1, \ldots, n\} \times \{1, \ldots, n\})$ where $q \in I$.
(c) $\widetilde{F}$ is the set of triples $(q, S, E) \in \widetilde{Q}$ such that $q \in F$.
(d) $\widetilde{\Delta}$ is the set of transitions $(q', S', E') \xrightarrow{[\![\Phi^S_{\widetilde{E},\widetilde{D}} \wedge \Psi]\!]} (q, S, \widetilde{E})$ such that

$$\left.\begin{array}{ll} q' \xrightarrow{[\![\Phi^S_{E,D} \wedge \Psi]\!]} q \text{ is in } \Delta & S' \supseteq S \\ & \widetilde{E} = E' \cap E \\ \Phi^S_{\widetilde{E},\widetilde{D}} \wedge \Psi \text{ satisfies conditions} & \widetilde{D} = E' \cap D \\ \text{i–iv of Proposition 2.4} & \end{array}\right\}. \tag{6}$$

Observe that the condition $E' \cap S^2 = \widetilde{E} \cup \widetilde{D}$ for oblivious automata holds, because the label $[\![\Phi^S_{\widetilde{E},\widetilde{D}} \wedge \Psi]\!]$ is an atom, so that $E \cup D = S^2$.

Let us denote by $\widetilde{T}$ the oblivious Σ_0-synchronous relation recognized by $\mathcal{O}$. In order to get i, we prove that $R = \widetilde{T}$, i.e., that $\widetilde{T}$ is the $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturation of T. Using i $\Rightarrow$ ii, we know that $\widetilde{T}$ is $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturated. So that it suffices to prove the two following properties:

(a) $T \subseteq \widetilde{T}$
(b) Every element of $\widetilde{T}$ is $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$ equivalent to some element of T.

Let us prove (a). We assign to every initial run $\rho_{\mathcal{A}}$ of $\mathcal{A}$ an initial run $\rho_{\mathcal{O}}$ of $\mathcal{O}$, such that $\rho_{\mathcal{A}}$ is successful if and only if so is $\rho_{\mathcal{O}}$ and such that the label of $\rho_{\mathcal{A}}$ is included in the label of $\rho_{\mathcal{O}}$. To this end, consider an initial run $\rho_{\mathcal{A}}$

$$q_0 \xrightarrow{[\![\Phi^{S_1}_{E_1,D_1} \wedge \Psi_1]\!]} q_1 \quad \ldots \quad q_{\ell-1} \xrightarrow{[\![\Phi^{S_\ell}_{E_\ell,D_\ell} \wedge \Psi_\ell]\!]} q_\ell \tag{7}$$

and assign it the run $\rho_{\mathcal{O}}$

$$(q_0, S_0, \widetilde{E_0}) \xrightarrow{[\![\Phi^{S_1}_{\widetilde{E_1},\widetilde{D_1}} \wedge \Psi_1]\!]} (q_1, S_1, \widetilde{E_1}) \quad \cdots\cdots \quad (q_{\ell-1}, S_{\ell-1}, \widetilde{E_{\ell-1}}) \xrightarrow{[\![\Phi^{S_\ell}_{\widetilde{E_\ell},\widetilde{D_\ell}} \wedge \Psi_\ell]\!]} (q_\ell, S_\ell, \widetilde{E_\ell}) \tag{8}$$

such that $S_0 = \{1, \ldots, n\}$, $\widetilde{E_0} = \{1, \ldots, n\} \times \{1, \ldots, n\}$ and, for $0 < k \leq \ell$ $\widetilde{E_k} = \widetilde{E_{k-1}} \cap E_{k-1}$ and $\widetilde{D_k} = \widetilde{E_{k-1}} \cap D_{k-1}$ holds. Observe that $(q_0, S_0, \widetilde{E_0})$ is initial in $\mathcal{O}$ and $(q_\ell, S_\ell, \widetilde{E_\ell})$ is final in $\mathcal{O}$ if and only if q_ℓ is final in $\mathcal{A}$. Let $(w_1, \ldots, w_n)$ be an n-tuple such that $\mathcal{H}(w_1, \ldots, w_n)$ belongs to the label of $\rho_{\mathcal{A}}$ and set $\ell = \max_{i=1,\ldots,n} |w_i|$. Then we have

$$\{(i, j) \mid w_i \restriction k = w_j \restriction k\} = \widetilde{E_k}$$
$$\{(i, j) \mid w_i \restriction (k-1) = w_j \restriction (k-1) \ \wedge \ w_i \restriction k \neq w_j \restriction k\} = \widetilde{D_k}$$

which means that $(w_1[k], \ldots, w_n[k])$ satisfies $[\![\Phi^{S_k}_{\widetilde{E_k},\widetilde{D_k}}]\!]$. This proves the inclusion claim (a).

As for property (b), observe that, due to the definition of $\mathcal{O}$, the n-tuples belonging to the label of an initial run of $\mathcal{O}$ are precisely the elements of a unique $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$ equivalence class of the label of an initial run of $\mathcal{A}$.

The implication i $\Rightarrow$ iv is trivial. We prove iv $\Rightarrow$ ii. As a preliminary observation, without loss of generality, we may assume that $\Sigma_2 = \Sigma_0 \subseteq \Sigma_1$. Indeed, since the relation is oblivious Σ_1-synchronous, it is a fortiori Σ_1-synchronous, thus by Theorem 3.5, it is $\Sigma_1 \cap \Sigma_2$-synchronous, i.e., Σ_0-synchronous. Let $\mathcal{A}$ be a Σ_0-synchronous automaton recognizing R. The hypothesis implies $|\Sigma \setminus \Sigma_1)| \geq n$. We prove that R is $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturated. Let $u = (u_1, \ldots, u_n)$ and $v = (v_1, \ldots, v_n)$ be in $(\Sigma^*)^n$. Supposing $u \in R$ and $u \equiv^{\mathrm{obl}}_{n,\Sigma_0} v$, we now show that $v \in R$.
For every $k \in \{1, \ldots, \max(|u_1|, \ldots, |u_n|)\}$ let X_k, respectively Y_k, be the letters in $\Sigma_1 \setminus \Sigma_0$ which occur in $\{u_i[k] \mid |u_i| \geq k\}$ and $\{v_i[k] \mid |v_i| \geq k\}$. Because of $|\Sigma \setminus \Sigma_1| \geq n$, there exist some permutations π_k and ρ_k of Σ which act as the identity on Σ_0 and which map X_k, respectively Y_k, into $\Sigma \setminus \Sigma_1$. Let $u' = (u'_1, \ldots, u'_n)$ be obtained from u by applying π_k on the letters of rank k, for each $k \in \{1, \ldots, \max(|u_1|, \ldots, |u_n|)\}$. Let $v' = (v'_1, \ldots, v'_n)$ be defined similarly with v and the ρ_k's.
Since the π_k's and ρ_k's act as the identity on Σ_0, we have (a) $u \equiv^{\mathrm{sync}}_{n,\Sigma_0} u'$ and (b) $v \equiv^{\mathrm{sync}}_{n,\Sigma_0} v'$. Since $u \equiv^{\mathrm{obl}}_{n,\Sigma_0} v$ and the equivalence $\equiv^{\mathrm{sync}}_{n,\Sigma_0}$ refines $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$, we deduce (c) $u' \equiv^{\mathrm{obl}}_{n,\Sigma_0} v'$. Now, u' and v' have no letter in $\Sigma_1 \setminus \Sigma_0$, hence (c) implies (d) $u' \equiv^{\mathrm{obl}}_{n,\Sigma_1} v'$.
Since $u \in R$ and R is $\equiv^{\mathrm{sync}}_{n,\Sigma_0}$-saturated, (a) implies $u' \in R$. Since R is $\equiv^{\mathrm{obl}}_{n,\Sigma_1}$-saturated, using (d) we get $v' \in R$. Using (b) and again the fact that R is $\equiv^{\mathrm{sync}}_{n,\Sigma_0}$-saturated, we finally obtain $v \in R$. □

Theorem 3.5 has an analog with oblivious synchronous relations.

Theorem 3.13. *Let Σ_1 and Σ_2 be two finite subsets of Σ such that $|\Sigma \setminus (\Sigma_1 \cup \Sigma_2)| \geq n$ (which is always the case if Σ is infinite). Then a relation $R \subseteq (\Sigma^*)^n$ is oblivious Σ_1 and oblivious Σ_2-synchronous if and only if it is oblivious $(\Sigma_1 \cap \Sigma_2)$-synchronous. In particular, if R is oblivious and if Σ is infinite then the smallest finite subalphabet $\Sigma_0 \subseteq \Sigma$ such that R is Σ_0-synchronous is also the smallest Σ_0 such that R is oblivious Σ_0-synchronous.*
Furthermore, this smallest subalphabet Σ_0 can be effectively computed.

Proof. Straightforward consequence of Theorems 3.5 and 3.12. □

Using condition ii in Theorem 3.12, we can extend the closure properties of EES relations (cf. Proposition 2.2) to synchronous relations.

Corollary 3.14. *Let Σ_1 and Σ_2 be two finite subsets of Σ, let R_1 be an oblivious Σ_1-synchronous relation and let R_2 be an oblivious Σ_2-synchronous relation. Let p be the projection defined by $p(w_1, \ldots, w_n) \mapsto (w_{i_1}, \ldots, w_{i_k})$ where n is the arity of R_1 and the i_j's are among $1, \ldots, n$.*
Then $p(R_1)$ and $(\Sigma^*)^n \setminus R_1$ are oblivious Σ_1-synchronous and $R_1 \times R_2$ is oblivious $(\Sigma_1 \cup \Sigma_2)$-synchronous. If R_1 and R_2 have the same arity then $R_1 \cup R_2$ and $R_1 \cap R_2$ are oblivious $(\Sigma_1 \cup \Sigma_2)$-synchronous.
Moreover, all these closure properties are effective in terms of oblivious synchronous automata.

Let us state a last decision property.

Theorem 3.15. *Let Σ is finite or infinite alphabet. There is an algorithm which, given two finite subalphabets Σ_0 and Σ_1 of Σ and a Σ_1-synchronous automaton $\mathcal{A}$, decides if the relation R recognized by $\mathcal{A}$ is oblivious Σ_0-synchronous.*

Proof. As in the proof of iv $\Rightarrow$ i in Theorem 3.12, from $\mathcal{A}$ we effectively construct an automaton $\mathcal{O}$ which recognizes the $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturated of R. Now, thanks to i $\Leftrightarrow$ ii from Theorem 3.4, the relation R is oblivious Σ_0-synchronous if and only if the two automata $\mathcal{A}$ and $\mathcal{B}$ recognize the same relation. □

4. Logics around synchronous relations

4.1. The main logics

The relations considered in [9] are those which are first-order definable in the following structure

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a \in \Sigma} \rangle$$

where Σ is a finite alphabet with at least two letters. The authors prove that they are identical with the relations recognized by (what is now called) synchronous automata. They observe that this result cannot be extended neither for one-letter alphabets nor for infinite ones: in both cases, the automata are more powerful than the logic. Here, we investigate the case of possibly infinite alphabets and consider, for every finite subalphabet Σ_0 of Σ, the structure

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a \in \Sigma_0} \rangle. \tag{9}$$

It turns out that the structure obtained by adding the predicate EqLenEqLast (which was considered by Eilenberg & al. in [9, Section 10, Problem 1]) is a crucial one:

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a \in \Sigma_0} \rangle. \tag{10}$$

We shall characterize the relations which are definable in structure (10) as the Σ_0-synchronous ones (cf. Theorem 4.1). In case $\Sigma_0 \neq \emptyset$, we characterize the relations which are definable in structure (9) as the oblivious Σ_0-synchronous ones (cf. Theorem 4.3). For the case $\Sigma_0 = \emptyset$, we introduce one more structure with no constant:

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{mod}_{k,\ell})_{k<\ell} \rangle \tag{11}$$

and we characterize the relations which are definable in structure (11) as the constant-free oblivious synchronous ones (cf. Theorem 4.5).
Also, we characterize relations which are definable in structure $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen} \rangle$ (i.e., structure (9) when $\Sigma_0 = \emptyset$) as those recognized by non-counting constant-free oblivious synchronous automata (cf. Theorem 4.8).
All these characterizations hold no matter whether the alphabet Σ is finite or infinite.

4.2. Encoding runs

We start with general observations concerning the encoding of a run of a finite automaton, which hold for the synchronous case, and the oblivious synchronous case with at least one constant. Consider such an automaton with N states and set $p = \lceil \log_2 N \rceil$ so that each state can be encoded as a length p sequence of 0's and 1's. Assume the alphabet of constants Σ_0 is nonempty, and contains an element, say $a \in \Sigma_0$. It is possible to encode each state of the automaton non uniquely, as a p-tuple $\sigma = (\sigma_1, \ldots, \sigma_p) \in \Sigma^p$ where $\sigma_i = a$ and $\sigma_i \neq a$ encode 0 and 1. Denote by $\langle \sigma \rangle$ the state encoded in such a way. A sequence of ℓ states can therefore be encoded as a p-tuple $(z_1, \ldots, z_p)$ of words of length ℓ. For $0 < i \leq \ell$, the i-th state of the sequence is encoded as the p-tuple consisting of the i-th letters of all z_j, $j = 1, \ldots, p$. Access to these letters is granted by the predicate $\mathtt{Last}_a$. Observe that in presence of the predicate $\mathtt{EqLenEqLast}$ we can do without constants. Indeed, add a $p+1$-th variable z_{p+1} to the previous p-tuple $(z_1, \ldots, z_p)$ and encode the sequence of states as follows: for each $1 \leq j \leq p$, the j-th binary digit of (a simple encoding of) the i-th state of the sequence is 1 if and only if *the prefixes of z_j and z_{p+1} of length i end with the same letter*.

To logically express the relation recognized by a finite automaton, we formalize the following property:

> Let $\overline{w} = (w_1, \ldots, w_n)$ be an n-tuple of words and let ℓ be their maximum length. The automaton recognizes $\overline{w}$ if there exists a sequence $q_0, q_1, \ldots, q_\ell$ of $\ell + 1$ states such that, for each position $0 < j \leq \ell$, the $n + 2$-tuple $(q_{j-1}, a_1, \ldots, a_n, q_j)$ belongs to a transition of the automaton, where for each i, either $|w_i| < j$ and $a_i = \#$ or a_i is the j-th letter of the input w_i. Furthermore, q_ℓ is a final state and q_0 is an initial state.

The technical translation below is probably better understood if the reader has in mind that the sequence of states is encoded with the p-tuple $(z_1, \ldots, z_p)$ and the position of the heads is encoded with the variable x via its length.

To this end we introduce the following functions and predicates on words whose purpose should be clear. Fixing some automaton, it is easy to check that all but the last one are definable from $\mathtt{Pref}$, $\mathtt{EqLen}$ and $\mathtt{Last}_a$ (where a is an arbitrary but fixed particular letter in Σ_0). As for the last one, it requires $\mathtt{Pref}$, $\mathtt{EqLen}$ et all $\mathtt{Last}_b$'s, $b \in \Sigma_0$.

- $\mathtt{Pref}_{|x|}(y)$ is the prefix of y of length $|x|$ and, more generally, $\mathtt{Pref}_{|x|+k}(y)$ is the prefix of y of length $|x| + k$ for any fixed integer $k \in \mathbb{Z}$.
- $\mathtt{State}_r(z_1, \ldots, z_p)$ is true if r is a state and $z_1, \ldots, z_p$ are p words of the same length such that r is encoded by the p-tuple of their last letters, more precisely, by the p-tuple of Boolean values $(\mathtt{Last}_a(z_1), \ldots, \mathtt{Last}_a(z_p))$.
- $\mathtt{SeqOfStates}(z_1, \ldots, z_p)$ is true if and only if $z_1, \ldots, z_p$ are words with the same length, such that the p-tuple of their first letters encodes an initial state, the p-tuple of their last letters encodes a final state and the p-tuple of the letters in any intermediate position encodes an arbitrary state.
- $\mathtt{Label}_\Phi$ is a predicate for each label of a transition. It is defined in the course of the proof.

4.3. The theory of $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a\in\Sigma_0} \rangle$

The following theorem states that the Σ_0-synchronous relations are exactly those which are definable in the above structure. It is worthwhile observing that the theorem is valid whether the alphabet of constants is empty or not, which will not be the case for the analogous result with oblivious synchronous automata and the theory without the predicate $\mathtt{EqLenEqLast}$.

Theorem 4.1. *Let Σ be a finite or infinite alphabet and let Σ_0 be a finite subalphabet of Σ. Then a relation $R \subseteq (\Sigma^*)^n$ is Σ_0-synchronous if and only if it is definable in the structure*

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a\in\Sigma_0} \rangle.$$

Observe that this solves Problem 2 for infinite alphabets. As originally formulated, this question has a negative answer since the predicate $\mathtt{EqLenEqLast}$ is EES and is invariant under all permutations but, as will be shown in Corollary 4.4, is not definable with $\mathtt{Pref}$, $\mathtt{EqLen}$ and the $\mathtt{Last}_a$'s. The present theorem insures a positive answer when we add this predicate $\mathtt{EqLenEqLast}$ to the logic.

Proof. "If" part. The proof is by induction on the complexity of formulae.
Atomic formulae. The relations defined by the atomic predicates can be recognized by synchronous automata. Indeed, Example 3.10 exhibits synchronous (even oblivious synchronous but for the first one), automata which recognize the relations $\mathtt{EqLenEqLast}$, $\mathtt{Last}_a$, $\mathtt{EqLen}$ and $\mathtt{Pref}$. This still holds when we identify variables in the above binary predicates.
Structural induction. It suffices to show that Σ_0-synchronous relations are closed under Boolean operations, projections and Cartesian product. But, this is a straightforward consequence of Corollary 3.6.

"Only if" part (under the hypothesis $\Sigma_0 \neq \emptyset$). The following formula expresses, in the language of our logic, that there exists a successful run labeled by the n-tuple $(w_1, \ldots, w_n)$.

$$\left.\begin{array}{l}\exists z_1 \ldots \exists z_p \, \exists z \, (|z_1| = \cdots = |z_p| = |z| \ \wedge \ |z| = 1 + \max_{1\le j\le n} |w_j|) \\ \wedge \ \mathtt{SeqOfStates}(z_1, \ldots, z_p) \\ \wedge \ \forall x \, (1 < |x| \le |z| \ \Rightarrow \\ \bigvee\limits_{\substack{(r, [\![\Phi]\!], s) \\ \text{is a transition}}} \begin{pmatrix} (\mathtt{State}_r(\mathtt{Pref}_{|x|-1}(z_1), \ldots, \mathtt{Pref}_{|x|-1}(z_p)) \\ \wedge \ \mathtt{State}_s(\mathtt{Pref}_{|x|}(z_1), \ldots, \mathtt{Pref}_{|x|}(z_p)) \\ \wedge \ \mathtt{Label}_{\Phi}(x, w_1, \ldots, w_n)) \end{pmatrix}\end{array}\right\} \tag{12}$$

where $\Phi = \Phi^S_{E,D} \wedge \Psi$ is as in Proposition 2.4 and $\mathtt{Label}_\Phi = \Theta_{\mathrm{E}} \wedge \Theta_{\mathrm{D}} \wedge \Theta_{\mathrm{S}} \wedge \Theta'$ with

$$\Theta_{\mathrm{E}} \ : \ \bigwedge_{(i,j)\in E} \mathtt{EqLenEqLast}(\mathtt{Pref}_{|x|}(w_i), \mathtt{Pref}_{|x|}(w_j)) \tag{13}$$

$$\Theta_{\mathrm{D}} \ : \ \bigwedge_{(i,j)\in D} \neg\mathtt{EqLenEqLast}(\mathtt{Pref}_{|x|}(w_i), \mathtt{Pref}_{|x|}(w_j)) \tag{14}$$

$$\Theta_{\mathrm{S}} \ : \ \bigwedge_{j\notin S} \big(|w_j| < |x|\big) \wedge \bigwedge_{j\in S} \big(|x| \le |w_j|\big) \, . \tag{15}$$

Furthermore, Θ' is obtained from Ψ by substituting $\mathtt{Last}_a(\mathtt{Pref}_{|x|}(w_i))$ for each occurrence of $x_i = a$ and $\neg\mathtt{Last}_a(\mathtt{Pref}_{|x|}(w_i))$ for each occurrence of $x_i \neq a$. "Only if" part (under the hypothesis $\Sigma_0 = \emptyset$). Encode the sequence of states with $p + 1$ words and use the predicate $\mathtt{EqLenEqLast}$ as explained at the beginning of Section 4.2. $\square$

As an easy consequence of Theorems 4.1 and 3.5, we have

Corollary 4.2. *Let Σ be an infinite alphabet. Given a relation definable in*

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a\in\Sigma} \rangle$$

there exists a unique smallest finite subset $\Sigma_0 \subseteq \Sigma$ such that Γ is definable in

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a\in\Sigma_0} \rangle.$$

4.4. *The theory of $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a\in\Sigma_0} \rangle$, where $\Sigma_0 \neq \emptyset$*

Here we solve Problem 1 and 3bis for oblivious Σ_0-synchronous automata in the case $\Sigma_0 \neq \emptyset$. The following result is the desired analog of Theorem 4.1 for oblivious synchronous relations.

Theorem 4.3. *Let Σ be a finite or infinite alphabet and let $\Sigma_0 \neq \emptyset$ be a finite subalphabet of Σ. A relation $R \subseteq (\Sigma^*)^n$ is oblivious Σ_0-synchronous if and only if it is definable in the structure $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a\in\Sigma_0} \rangle$.*

Proof. "Only if part". The formula is as in (12) in the proof of Theorem 4.1, except for the interpretation of expressions (13) and (14) in which, using the oblivious character of the automaton, we replace the $\mathtt{EqLenEqLast}$ predicate as follows:

$$\Theta_{\mathrm{E}} \ : \ \bigwedge_{(i,j)\in E} \mathtt{Pref}_{|x|}(w_i) = \mathtt{Pref}_{|x|}(w_j)$$

$$\Theta_{\mathrm{D}} \ : \ \bigwedge_{(i,j)\in D} (\mathtt{Pref}_{|x|}(w_i) \neq \mathtt{Pref}_{|x|}(w_j)) \wedge \ (\mathtt{Pref}_{|x|-1}(w_i) = \mathtt{Pref}_{|x|-1}(w_j)).$$

"If part". The proof proceeds as in the previous theorem since the family of oblivious relations enjoys the same closure properties. $\square$

The solution to Problem 1 and Problem 3bis is a consequence of the above results. Actually, we obtain a negative answer to Problem 1 and a positive answer to Problem 3bis in case Σ_0 is nonempty.

Corollary 4.4. *Let Σ be an infinite alphabet.*
1. *Neither* $\mathtt{EqLenEqLast}$ *nor* $\{(xz, yz) \mid x, y, z \in \Sigma\}$ *(i.e., the restriction of* $\mathtt{EqLenEqLast}$ *to words of length* 2*) are definable with* $\mathtt{Pref}$, $\mathtt{EqLen}$ *and the* $\mathtt{Last}_a$*'s for* $a \in \Sigma$.

2. *Suppose $\Sigma_0 \neq \emptyset$. If a relation $R \subseteq (\Sigma^*)^n$ is definable with* $\mathtt{Pref}$, $\mathtt{EqLen}$ *and the* $\mathtt{Last}_a$*'s and is invariant under all permutations which are the identity on Σ_0 then it is definable with* $\mathtt{Pref}$, $\mathtt{EqLen}$ *and the* $\mathtt{Last}_a$*'s for $a \in \Sigma_0$.*

Of course, if Σ is finite then $\mathtt{EqLenEqLast}$ is trivially definable with $\mathtt{EqLen}$ and all the $\mathtt{Last}_a$'s. Also, it is not definable with the sole $\mathtt{Last}_a$'s, $a \in \Sigma_0$, if $|\Sigma \setminus \Sigma_0| \geq 2$, though it is invariant under all permutations.

Proof. 1. It suffices to prove the result with $R = \{(xz, yz) \mid x, y, z \in \Sigma\}$. Suppose R were definable with $\mathtt{Pref}$, $\mathtt{EqLen}$ and the $\mathtt{Last}_a$'s. Let Σ_0 be the finite non empty set consisting of all a's such that $\mathtt{Last}_a$ occurs in a formula defining R. Theorem 4.3 insures that R is recognizable by some oblivious Σ_0-synchronous automaton. Now, if b, c are distinct letters in $\Sigma \setminus \Sigma_0$ then we have $(bc, cb) \equiv^{\mathrm{obl}}_{n,\Sigma_0} (bb, cb)$ but $(bb, cb) \in R$ and $(bc, cb) \notin R$.

2. By Theorem 4.3, R is oblivious Σ_1-synchronous for some finite subalphabet Σ_1 and by Theorem 3.4 it is Σ_0-synchronous. Finally, Theorem 3.12 insures that R is oblivious Σ_0-synchronous. $\square$

4.5. The theory of $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{mod}_{k,\ell})_{k<\ell}\rangle$

The logical characterization of Σ_0-synchronous relations stated in Theorem 4.1 is valid whatever be the finite set Σ_0. On the opposite, Theorem 4.3 characterizes oblivious Σ_0-synchronous relations when $\Sigma_0 \neq \emptyset$. It turns out that, when $\Sigma_0 = \emptyset$, we have to use the extra modular predicates $\mathtt{mod}_{k,\ell}$ (meaning that "*the length of u is congruent to k modulo ℓ*"). In fact, the $\mathtt{mod}_{k,\ell}$ predicates are easy to define with $\mathtt{Pref}$ and $\mathtt{Last}_a$ for any fixed $a \in \Sigma$: $\mathtt{mod}_{k,\ell}(u)$ is true if and only if there exists some v of the same length as u, ending with the letter a and having an occurrence of the letter a in exactly the positions equal to k modulo ℓ. However, no predicate $\mathtt{mod}_{k,\ell}$ with $k < \ell$ is definable without constants, cf. Theorem 4.8.

The analog of Theorem 4.3 when $\Sigma_0 = \emptyset$ is as follows.

Theorem 4.5. *Let Σ be a finite or infinite alphabet and let $R \subseteq (\Sigma^*)^n$. Then R is oblivious constant-free synchronous if and only if it is definable in the structure $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{mod}_{k,\ell})_{k<\ell}\rangle$.*

Proof. "If" part. The proof of Theorem 4.1 carries over to this case, by using the fact that $\mathtt{mod}_{k,\ell}$ is oblivious constant-free synchronous (cf. Example 3.10, Fig. 5).

"Only if" part. Let's prove that the relation recognized by a constant-free oblivious synchronous automaton $\mathcal{O}$ can be expressed by some formula of the logic. In the present case, since there is no constant, transitions are labeled by expressions $[\![\Phi]\!]$ where Φ is of the form

$$\bigwedge_{i \notin S_\Phi}(x_i = \#) \ \wedge\ \bigwedge_{i \in S_\Phi}(x_i \neq \#) \ \wedge\ \bigwedge_{(i,j)\in E_\Phi}(x_i = x_j) \ \wedge\ \bigwedge_{(i,j)\in D_\Phi}(x_i \neq x_j).$$

Given two such expressions Φ' and Φ we write $[\![\Phi']\!] \succeq [\![\Phi]\!]$ whenever the following conditions (which are those for transitions of oblivious automata, cf. (5)) hold:

$$S_\Phi \subseteq S'_\Phi, \qquad E_\Phi \subseteq E'_\Phi, \qquad E'_\Phi \cap (S_\Phi \times S_\Phi) = E_\Phi \cup D_\Phi.$$

If, furthermore $[\![\Phi']\!] \neq [\![\Phi]\!]$ holds, we write $[\![\Phi']\!] \succ [\![\Phi]\!]$. Observe that if $[\![\Phi']\!] \succeq [\![\Phi]\!]$ and $S_\Phi = S'_\Phi$ and $E_\Phi = E'_\Phi$ then $D_\Phi = \emptyset$.

Every run of $\mathcal{O}$ is of the form

$$(q_0, S_0, E_0) \xrightarrow{[\![\Phi'_1]\!]} (q_1, S_1, E_1) \xrightarrow{[\![\Phi'_2]\!]} \cdots (q_{t-1}, S_{t-1}, E_{t-1}) \xrightarrow{[\![\Phi'_t]\!]} (q_t, S_t, E_t)$$

where for some sequence $0 < s_1 < s_2 \cdots < s_{p-1} < s_p = t$ we have

$$\begin{array}{llllll}
\Phi'_1 = \cdots = \Phi'_{s_1} & \succ \Phi'_{s_1+1} & = \cdots = \Phi'_{s_2} & \succ \cdots \succ \Phi'_{s_{p-1}+1} & = \cdots = \Phi'_t \\
S_{\Phi'_1} = \cdots = S_{\Phi'_{s_1}} & \supseteq S_{\Phi'_{s_1+1}} & = \cdots = S_{\Phi'_{s_2}} & \supseteq \cdots \supseteq S_{\Phi'_{s_{p-1}+1}} & = \cdots = S_{\Phi'_t} \\
E_{\Phi'_1} = \cdots = E_{\Phi'_{s_1}} & \supseteq E_{\Phi'_{s_1+1}} & = \cdots = E_{\Phi'_{s_2}} & \supseteq \cdots \supseteq E_{\Phi'_{s_{p-1}+1}} & = \cdots = E_{\Phi'_t}.
\end{array}$$

Observe that the integer p is bounded by $2n-2$ (where n is the arity of the automaton). Indeed, associate with Φ the integer $2 \times |S_\Phi| - c_\Phi$ where c_Φ is the number of equivalence classes of E_Φ. Then the maximum value of this quantity is $2n-1$ and its minimum value is 1. Also, this quantity strictly decreases when going from Φ' to Φ such that $\Phi' \succ \Phi$.

Set $r_1 = s_1$, $r_{i+1} = s_{i+1} - s_i$ for $0 < i < p$ and $\Phi_i = \Phi'_{s_i}$ for $0 < i \leq p$. Observe that $r_i = 1$ if the formula Φ_i contains an inequality, i.e., $D_{\Phi_i} \neq \emptyset$. Also, $[\![\Phi_1]\!] \succ [\![\Phi_2]\!] \cdots \succ [\![\Phi_p]\!]$ and the label of the above run is $[\![\Phi_1]\!]^{r_1}[\![\Phi_2]\!]^{r_2} \ldots [\![\Phi_p]\!]^{r_p}$.

Now, fix the states (q_0, S_0, E_0), $(q_{s_1}, S_{s_1}, E_{s_1}) \ldots (q_{s_p}, S_{s_p}, E_{s_p})$. The possible values of the exponents $r_1, r_2, \ldots, r_p$ belong to a rational subset of $\mathbb{N}$, i.e., the subset of labels obtained by letting the r_i's vary is of the form

$$[\![\Phi_1]\!]^{K_1}[\![\Phi_2]\!]^{K_2} \ldots [\![\Phi_p]\!]^{K_p} \tag{16}$$

where the K_i's are recognizable subsets $\mathbb{N}$ (i.e., ultimately periodic subset of integers). Since the automaton is finite, for each integer $0 < p \leq 2n-2$ there are only finitely many ways of fixing the $p+1$ states $(q_{s_k}, S_{s_k}, E_{s_k})$, $k = 0, \ldots, p$. Therefore, the relation recognized by $\mathcal{O}$ is a finite union of subsets as in expression (16).

Now, the formula $\phi(w_1, \ldots, w_n)$ defining the relation (16) is of the form

$$\exists y_1 \exists y_2 \ldots \exists y_p \quad \psi_1(y_1, \ldots, y_p) \wedge \psi_2(w_1, \ldots, w_n, y_1, \ldots, y_p) \wedge \psi_3(w_1, \ldots, w_n, y_1, \ldots, y_p) \tag{17}$$

where

- $|y_k|$ is the maximum length of a component of the prefix of $(w_1, \ldots, w_n)$ which is in $[\![\Phi_1]\!]^{K_1} \cdots [\![\Phi_k]\!]^{K_k}$,
- formulae ψ_1, ψ_2, ψ_3 are used to express the different lengths of the variables $w_1, \ldots, w_n$ and of their pairwise maximum common prefixes.

Formally, for $i, j = 1, \ldots, n$, let λ_i be the largest index $1 \le \lambda \le p$ such that $i \in S_{\Phi_\lambda}$ and let $\mu_{i,j}$ be the largest integer $0 \le \mu \le p$ such that $(i, j) \in E_{\Phi_\mu}$ holds. Formulae ψ_1, ψ_2, ψ_3 are as follows

$$\begin{array}{rl} \psi_1(y_1, y_2, \ldots, y_p): & (|y_1| \in K_1) \wedge (|y_2| - |y_1| \in K_2) \ldots \wedge (|y_p| - |y_{p-1}| \in K_p) \\ \psi_2(w_1, \ldots, w_n): & \bigwedge_{1 \le i \le n} |w_i| = |y_{\lambda_i}| \\ \psi_3(w_1, \ldots, w_n): & \bigwedge_{1 \le i,j \le n} |y_{\mu_{i,j}}| = |\,\mathrm{MCP}(w_i, w_j)| \end{array}$$

where the function $\mathrm{MCP}(u, v)$ (which maps (u, v) to their maximum common prefix) is definable with $\mathtt{Pref}$, and, using the $\mathtt{EqLen}$ and $\mathtt{mod}_{k,\ell}$ predicates, we can also define the predicates $|u| - |v| = k \bmod \ell$ and $|u| - |v| \in K$ for any recognizable subset K of $\mathbb{N}$. $\square$

The above proof can be used to prove elimination of quantifiers.

Proposition 4.6. *Let Σ be a finite or infinite alphabet. The theory of the structure $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{mod}_{k,\ell})_{k<\ell} \rangle$ admits elimination of quantifiers in the language consisting of*
- a predicate $|x| - |y| \in K$ for each recognizable subset K of $\mathbb{N}$,
- a constant ε representing the empty word,
- a function $\mathrm{MCP}(x, y)$ representing the maximal common prefix of x and y.

Proof. To eliminate the existential quantifications in formula (17), it suffices to prove that, for each variable y_k in (17), there exists a pair (w_i, w_j) such that $|y_k| = |\mathrm{MCP}(w_i, w_j)|$. In fact, $|y_p| = |w_i| = |\mathrm{MCP}(w_i, w_i)|$ for any $i \in S_{\Phi_p}$ (recall $\Phi_i = \Phi'_{s_i}$). For $k < p$, either $S_{\Phi_k} \supsetneq S_{\Phi_{k+1}}$ or $E_{\Phi_k} \supsetneq E_{\Phi_{k+1}}$. In the first case, $|y_k| = |\mathrm{MCP}(w_i, w_i)|$ for any $i \in S_{\Phi_k} \setminus S_{\Phi_{k+1}}$. In the second case, $|y_k| = |\mathrm{MCP}(w_i, w_j)|$ for any $(i, j) \in E_{\Phi_k} \setminus E_{\Phi_{k+1}}$.
Finally, observe that $\mathtt{Pref}$ can be expressed as $|\mathrm{MCP}(x, y)| - |\mathrm{MCP}(x, x)| \ge 0$. $\square$

As a corollary we can solve Problem 3.

Corollary 4.7. *Let Σ be an infinite alphabet. If a relation is definable with $\mathtt{Pref}$, $\mathtt{EqLen}$ and the $\mathtt{Last}_a$'s and is invariant under all permutations then*

(1) *it is definable with $\mathtt{Pref}$, $\mathtt{EqLen}$ and the $\mathtt{mod}_{k,\ell}$'s for $k < \ell$*
(2) *it is definable with $\mathtt{Pref}$, $\mathtt{EqLen}$ and only one predicate $\mathtt{Last}_a$, where a is any letter in Σ.*

The result fails when Σ is finite, cf. the counterexample given after the statement of Corollary 4.4.

Proof. 1. By Theorem 4.3, R is oblivious Σ_1-synchronous for some finite subalphabet Σ_1. Now, by Theorem 3.4, R is constant-free synchronous. Finally, Theorem 3.12 shows that that R is oblivious constant-free synchronous.
2. Apply point 2 of Corollary 4.4 with $\Sigma_0 = \{a\}$. $\square$

4.6. The theory of $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen} \rangle$,

We now deal with the original structure considered in [9] via the notion of non-counting automaton. An automaton is *non-counting* when all shortest non empty runs taking some state to itself have length equal to 1. We shall say that a relation is non-counting EES if it is recognized by some non-counting EES automaton. Similarly, we shall speak of non-counting synchronous and non-counting oblivious synchronous.

Theorem 4.8. *Let Σ be a finite or infinite alphabet and let $R \subseteq (\Sigma^*)^n$. Then R is non-counting oblivious constant-free synchronous if and only if it is definable in the structure $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen} \rangle$.*

Proof. Here again the arguments are direct adaptations of those used in the proof of Theorem 4.5 by using the following observations. For the "only if" direction, observe that since $\mathcal{O}$ is non-counting, the K_i's are either finite or cofinite and $|u| - |v| \in K_i$ is expressible with $\mathtt{Pref}$ and $\mathtt{EqLen}$.

Conversely, observe that the automata for $\mathtt{Pref}$ and $\mathtt{EqLen}$ in Example 3.10 are non-counting and that non-counting EES relations are closed under projections, Cartesian product and Boolean operations. $\square$

We also have a simple elimination of quantifiers.

Proposition 4.9. *Let Σ be a finite or infinite alphabet. The theory of the structure $\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen} \rangle$ admits elimination of quantifiers in the language consisting of*
- a predicate $|x| - |y| \ge \ell$ for each $\ell \in \mathbb{N}$,
- a constant ε and a function $MCP(x, y)$ as in Proposition 4.6.

4.7. Oblivious relations versus "regular prefix relations"

The notion of oblivious synchronous relation has some similarity with that of "regular prefix relation" introduced by Angluin and Hoover, 1984 [1]. We now show that the two notions, considered for a finite alphabet Σ, are in fact incomparable.

Recall that regular prefix relations on Σ^* constitute the smallest class of relations containing all regular languages and such that, if $R, R_1, \ldots, R_k$ are regular prefix relations then so are the cartesian product $R_1 \times \cdots \times R_k$ and the concatenation product $\{\overbrace{(u, \ldots, u)}^{n \text{ times}} \mid u \in L\}\, R$ and $\theta(R)$ where θ is a permutation of the components and L is a regular language and n is the arity of R.

Let Σ be a finite alphabet. Consider the two structures

$$\langle \Sigma^*; =, (u \mapsto ua)_{a\in\Sigma}\rangle, \quad \langle \Sigma^*; \varepsilon, \mathrm{MCP}, (P_L)_{L\in\mathrm{Reg}(\Sigma^*)}\rangle$$

where MCP is as in Proposition 4.9, $\mathrm{Reg}(\Sigma^*)$ is the class of regular languages included in Σ^* and $P_L = \{(u, v) \mid v \in uL\}$. Laüchli and Savioz, 1987 [11], proved that a relation on Σ^* is regular prefix if and only if it is definable by some monadic second-order formula (with all second-order variables bounded) in the first structure if and only if it is first-order definable in the second one (cf. Choffrut [6]).

Case $|\Sigma \setminus \Sigma_0| \le 1$. Observe that any letter in a finite alphabet Σ is definable from the other ones: $\mathtt{Last}_a(u) \Leftrightarrow \bigwedge_{b\neq a} \neg\mathtt{Last}_b(u)$. Thus, in the considered case, Σ_0-synchronous, oblivious Σ_0-synchronous and synchronous regular (in the usual sense) are the same notion which encompasses that of regular prefix.

Case $|\Sigma \setminus \Sigma_0| \ge 2$. Then the class of regular prefix relations is not comparable neither with that of Σ_0-synchronous relations nor that of oblivious Σ_0-synchronous relations.

Indeed, EqLen is oblivious constant-free synchronous but not regular prefix.

Also, though P_L is Σ_0-synchronous for every regular language $L \subseteq \Sigma_0^*$, there are regular languages L such that P_L is not Σ_0-synchronous (a fortiori not oblivious Σ_0-synchronous) for any subalphabet Σ_0 such that $|\Sigma \setminus \Sigma_0| \ge 2$. For instance, suppose $\Sigma = \{a, b, c\}$, $\Sigma_0 = \{a\}$ and $L = a^* \cup b^* \cup c^*$. The smallest relation containing P_L and invariant under all level-by-level permutations of Σ which, at every level, leave a fixed (cf. the definition given before Theorem 3.4) is $P_{a^*\cup\{b,c\}^*}$, which properly contains P_L. Using condition iv in Theorem 3.4, we see that P_L is not Σ_0-synchronous.

4.8. Modular quantifiers

In this paragraph we show that modular quantifiers do not increase the expressive power of the two main logics. Recall that

- $\exists^\infty w_n\, \phi(w_1, \ldots, w_n)$ is true if there exists infinitely many values for the variable w_n for which the expression is true,
- $\exists^{k \bmod \ell} w_n\, \phi(w_1, \ldots, w_n)$ (where $0 \le k < \ell$) is true if the number of values of w_n such that the expression is true (for given $w_1, \ldots, w_{n-1}$) is finite and congruent to k modulo ℓ.

Theorem 4.10. *Let Σ be a finite or infinite alphabet. There is an algorithm which, given a finite subalphabet Σ_0 and a formula of the language*

$$(\mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a\in\Sigma_0}) \text{ (resp. } \mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a\in\Sigma_0})$$

using quantifiers $\exists$, $\exists^\infty$ and $\exists^{k \bmod \ell}$ (all pairs $k < \ell$), associate a Σ_0-synchronous (resp. oblivious Σ_0-synchronous) automaton recognizing the relation defined by the given formula.

Proof. Observe that, for $n \ge 2$, if $R \subseteq (\Sigma^*)^n$ is $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturated then so is the relation $\{(w_1, \ldots, w_{n-1}) \mid Q w_n\, (w_1, \ldots, w_{n-1}, w_n) \in R\}$ where Q is any of the quantifiers $\exists$, $\exists^\infty$ or $\exists^{k \bmod \ell}$. Also, $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturation is preserved by boolean operations and all relations Pref, EqLen and $\mathtt{Last}_a$, for $a \in \Sigma_0$, are $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturated. This insures that every relation definable in the language $(\mathtt{Pref}, \mathtt{EqLen}, (\mathtt{Last}_a)_{a\in\Sigma_0})$, using the mentioned generalized quantifiers is $\equiv^{\mathrm{obl}}_{n,\Sigma_0}$-saturated. In particular, using Theorem 3.12, this shows that the statement of the theorem relative to the first structure yields that relative to the second one.

We shall use variables $w_1, \ldots, w_n$ to vary over words in Σ^* in formula ϕ and variables $x_1, \ldots, x_n$ to vary over letters in Σ in formulae Φ, Ψ.

The intuition of the proof of the result about the first structure is as follows. Consider the set $\overline{\mathbb{N}}_\ell = \{0, \ldots, \ell - 1, \infty\}$ with modular addition and multiplication on $\{0, \ldots, \ell - 1\}$ extended as follows:

$$\forall \xi \in \overline{\mathbb{N}}_\ell \;\; \infty + \xi = \xi + \infty = \infty\xi = \xi\infty = \infty.$$

In all this paragraph the computations are meant in these laws. Given an n-tape Σ_0-synchronous automaton $\mathcal{A}$ recognizing the relation R defined by the formula ϕ, we construct an automaton $\mathcal{B}$, which recognizes the relation defined by the formulae $\exists^\infty w_n \phi(w_1, \ldots, w_n)$ (resp. $\exists^{k \bmod \ell} w_n \phi(w_1, \ldots, w_n)$) by specifying which states are final. To that purpose we perform a subset construction which carries more information than the standard one. Indeed, along with each state p of a superstate

$P \subseteq Q$ we record the number in $\overline{\mathbb{N}}_\ell$ of runs taking the initial state to p and having the same projection on the $n-1$ first components: ∞ if it is infinite and k if it is finite and congruent to k modulo ℓ.

We start with a preliminary remark. We claim that it can be easily decided whether or not the subset of Σ^* recognized by some one-tape deterministic Σ_0-synchronous automaton $\mathcal{A}$ is infinite, and if this is not the case, whether its cardinality is equal to k modulo ℓ. Indeed, without loss of generality we can assume that $\mathcal{A}$ is trimmed (i.e., all states are accessible and final states are accessible from any state) and that each label is either a subset of Σ_0 or contains $\Sigma \setminus \Sigma_0$. Now, the subset recognized by the automaton is infinite either if there exists a loop or if there exists a successful run containing a transition labeled by some cofinite subset of Σ. If the language is finite, it suffices to compute its cardinality modulo ℓ.

We now turn to the proof of the theorem. Without loss of generality, we can suppose that $\mathcal{A}$ is trimmed and deterministic and that the labels of the transitions are atoms of the algebra $\mathfrak{F}^n_{\Sigma_0}$. Let Q be its set of states and F be its set of final states. With each state $q \in Q$ associate the value $\gamma(q) \in \overline{\mathbb{N}}_\ell$ which is the cardinality of the subset of Σ^* recognized by the automaton obtained from $\mathcal{A}$ by fixing q as initial state and by deleting all transitions whose labels have a support different from $\{n\}$. This function γ can be computed as explained in the preliminary claim.

The state set of $\mathcal{B}$ is the collection of all elements (P, β) where $P \subseteq Q$ and $\beta : P \to \overline{\mathbb{N}}_\ell$.
Given an atom $[\![\Phi]\!]$ of $\mathfrak{F}^{n-1}_{\Sigma_0}$, where $\Phi = \Phi^S_{E,D} \wedge \Psi$ is quantifier-free with variables $x_1, \ldots, x_{n-1}$ (cf. Proposition 2.4), we let $\mathcal{F}([\![\Phi]\!])$ be the family of atoms $[\![\Phi^+]\!]$ of $\mathfrak{F}^n_{\Sigma_0}$, where Φ^+ is quantifier-free with variables $x_1, \ldots, x_{n-1}, x_n$ such that $\forall x_n\, (\Phi^+(x_1, \ldots, x_n) \Rightarrow \Phi(x_1, \ldots, x_{n-1}))$ is true, i.e., $[\![\Phi^+ \Rightarrow \Phi]\!]$ is the top element of the Boolean algebra $\mathfrak{F}^n_{\Sigma_0}$.
Transitions of $\mathcal{B}$ with label $[\![\Phi]\!]$ are of the form

$$(P', \beta') \xrightarrow[\mathcal{B}]{[\![\Phi]\!]} (P, \beta)$$

where P is the set of elements $p \in Q$ for which there exists $p' \in P'$ such that

$$p' \xrightarrow[\mathcal{A}]{[\![\Phi^+]\!]} p \text{ is a transition in } \mathcal{A} \text{ with } [\![\Phi^+]\!] \in \mathcal{F}([\![\Phi]\!]) \qquad (18)$$

and $\beta(p) = \sum_{p',\, [\![\Phi^+]\!]} \lambda_{p',\Phi^+}\, \beta'(p')$ where

- $(p', [\![\Phi^+]\!])$ ranges over all possible pairs in $P' \times \mathcal{F}([\![\Phi]\!])$ satisfying condition (18),
- and $\lambda_{p',\Phi^+} = 1$ if Φ^+ implies $x_n = \#$ or $x_n = a$ for some $a \in \Sigma_0$ or $x_n = x_i$ for some $i \in \{1, \ldots, n-1\}$ and $\lambda_{p',\Phi^+} = \infty$ otherwise.

The initial state of $\mathcal{B}$ is the pair $(\{q_0\}, \beta_0)$ where q_0 is the initial state of $\mathcal{A}$ which we may assume without incoming transition and $\beta_0(q_0) = 1$ holds.

The cases of quantifiers $\exists^\infty$ and $\exists^{k \bmod \ell}$ differ solely in the choice of the final states of $\mathcal{B}$. A pair (P, β) is final if and only if the sum

$$\left(\sum_{p \in F \cap P,\, \gamma(p)=0} \beta(p) \right) + \left(\sum_{p \in P,\, \gamma(p) \neq 0} \beta(p)\gamma(p) \right)$$

is equal to ∞ in the case of the quantifier $\exists^\infty$ and to k mod ℓ in the case of the quantifier $\exists^{k \bmod \ell}$. □

4.9. A decidable theory

Theorem 4.11. *Let Σ be a finite or infinite alphabet. The first order theory of the structure*

$$\langle \Sigma^*; \mathtt{Pref}, \mathtt{EqLen}, \mathtt{EqLenEqLast}, (\mathtt{Last}_a)_{a \in \Sigma} \rangle$$

is decidable, even if quantifiers $\exists^\infty$ and $\exists^{k \bmod \ell}$ are allowed.

Proof. Observe that the emptiness problem for Σ_0-synchronous automata is trivially decidable in a uniform way with respect to the parameter Σ_0 varying among finite subsets of Σ. To conclude, use Theorems 4.1 and 4.10. □

References

[1] D. Angluin, D.N. Hoover, Regular prefix relations, Mathematical Systems Theory 17 (3) (1984) 167–191.
[2] M. Benedikt, L. Libkin, T. Schwentick, L. Ségoufin, Definable relations and first-order query languages over strings, Journal of the Association of Computing Machinery 50 (2003) 694–751.
[3] J. Berstel, Transductions and Context-Free Languages. B.G. Teubner, 1979.
[4] A. Bès, An application of the Feferman-Vaught theorem to automata and logics for words over an infinite alphabet, Logical Methods in Computer Science (in press).
[5] M. Bojanczyk, A. Muscholl, T. Schwentick, L. Segoufin, C. David, Two-variable logic on words with data, in: Proceedings of LICS'06, 2006, pp. 7–16.
[6] C. Choffrut, Relations over words and logic: A chronology, Bulletin of the EATCS 89 (June) (2006) 159–163.
[7] S. Demri, R. Lazić, A. Sangnier, Model checking freeze LTL over one-counter automata, in: Proceedings of FoSSaCS'08, in: LNCS, 2008 (in press).
[8] S. Eilenberg, Automata, Languages and Machines, vol. A, Academic Press, 1974.
[9] S. Eilenberg, C.C. Elgot, J.C. Shepherdson, Sets recognized by n-tape automata, Journal of Algebra 13 (1969) 447–464.
[10] D. Epstein, J. Cannon, D.F. Holt, S.V.F. Levy, M.S. Paterson, W.P. Thurston, Word Processing in Groups, Jones and Bartlett, 1992.
[11] H. Läuchli, C. Savioz, Monadic second order definable relations on the binary tree, Journal of Symbolic Logic 52 (1) (1987) 219–226.
[12] J. Sakarovitch, Eléments de théorie des automates, Vuibert Informatique, 2003.
[13] W. Thomas, Infinite trees and automaton-definable relations over the ω-words, Theoretical Computer Science 103 (1992) 143–159.